robotron

Anwendungsbeschreibung der Dienst- und Hilfsprogramme für SIOS 1520

Bedienerhilfen

A 5120/30

Bedienerwortschatz englisch
für die Arbeit mit den Bürocomputern
A 5120/30

engl.	Abkürzung	deutsch
A		
above		über
account cart	AC	Kontokarte
address		Adresse
again		wieder
all		alles
already		bereits
and		und
area		Feld
assembling		Assemblierung
assign	ASSGN	Zuweisung
at		zu, zum
available		zulässig
B		
backward		rückwärts
binary		binär
blocked		geblockt

engl.	Abkürzung	deutsch
book		Buch
by		bei
C		
calculate		Berechnung
call	CAL	Ruf
can		können
cancel	CAN	ungültig machen
capazity		Kapazität
carriage		Wagen (Büromaschine)
cartridge	CT	Magnetbandkassette
cassette		Kassette
change		Veränderung, Wechsel
charakters		Zeichen (Bit)
check		Kontrolle
clear		klar
clock		Uhr
close		schließen
collate		mischen
command		Kommando
comparison		Vergleich
CPU		Zentrale Verarbeitungseinheit

engl.	Abkürzung	deutsch
copy		kopieren
correction		Korrektur
counter		Zähler
cut		Schalter
cycle		Zyklus
D		
data		Daten
decimal		dezimal
delete		löschen
destination		Bestimmung
destroyed		zerstört
device	DEV	~~Zuweisung~~ Gerät
dictionary	DIC	Adressenverzeichnis
digit		Dezimalstelle
directory		Verwaltung
disk		Scheibe
display	DLPLY	Bildschirm
drive		antreiben

engl.	Abkürzung	deutsch
E		
enlargement		Erweiterung
enter		eintreten
equal		gleich
erase		auslöschen
error		Fehler
extend		Ausdehnung
extent		ausdehnen, erweitern
F		
factory		Faktor
feed		Vorschub
file		Datei
filled		gefüllt
fin		Ende
first		erster; zuerst
floppy-disk	FD	Folienspeicher
for		für
forward		vorwärts
found		finden
from		von
full		voll

engl.	Abkürzung	deutsch
G		
generate		Generierung
go		gehen
H		
header	HDR	Kopf
help		Hilfe
I		
ident	ID	erkannt
impossible		unmöglich
incompatible		unvereinbar
initialize		initialisieren
input		Eingabe
insert		einfügen
intensity		Intensität
interrupt		Unterbrechung
invalid		defekt
is		ist

engl.	Abkürzung	deutsch
K		
Key		Schlüssel
Keyboard	KB	Tastatur
L		
label		Marke
last		letzter, zuletzt
length		Länge
library	L/LB	Bibliothek
line		Zeile
list		Liste
load		laden
logical		logisch
M		
marked		~~Marke~~ markiert
memory		Speicher
missing		fehlend

engl.	Abkürzung	deutsch
N		
name		Name
need		benötigt
new		neu
next		nächster, zunächst
no	N	nein
not		nicht
nothing		Null
number		Nummer
O		
of		von
on		an, bei
open		offen
option		Möglichkeit
or		oder
organization		Organisation
our		unser
output		Ausgabe

engl.	Abkürzung	deutsch
P		
page		Seite
paper		Papier
part		Abschnitt
partial		teilweise
pass		Durchlauf
permissible		gestatten
phase	P	Phase
physical		physisch
position		Stelle
printer	PR	Drucker
protect		schützen
punch		stanzen
put		einlegen
R		
read	R	lesen
record		Satz
reduction		Zurückführung, Herabsetzung
relocation		rückspeichern
remainder		Rest

engl.	Abkürzung	deutsch
rename		umbenennen
repeat		Wiederholung
reset		Rücksetzen
return		zurückkehren
rewind		rückspulen
S		
save		Schutz
searching		Suche
sector		Sektor
self		selbst
send		senden
sequence	SEQU	Reihenfolge, Ordnung
sign		Zeichen
size		Format
skip		Vorschubbewegung d.Papiers
sort		sortieren
source	S	Quelle
space		Zwischenraum
span		Spanne
specifikation		Spezifikation

engl.	Abkürzung	deutsch
stackpointer		Kellerzeiger
statement		Verzeichnis, Angabe
store		Vorrat, Speicher
structure		Struktur
sum		Summe
T		
tape		(Loch-) Band
the		der, die, das
time		Zeit
timer		Zeitgeber
to		nach
track		Spur
traverse		quer
two		zwei
U		
unit		Einheit, Gerät
unlike		ungleich
use		Gebrauch, Anwendung

engl.	Abkürzung	deutsch
V		
variance		Variante
Volume	VOL	Inhalt
W		
wait		warten
weigth		Gewicht
without		ohne
working		Arbeit
write	W	schreiben
Y		
yes	y	ja
your		~~einer~~ Euer
Z		
Zero		Null

Allgemeine Hinweise

Einzelne Wörter können durch Vor- oder Nachsilben Veränderungen unterliegen. Häufig anzutreffen sind:

die Vorsilben

in	ein	z.B.	input	Eingabe
re	rück-	z.B.	reset	rücksetzen

die Nachsilben:

ing
er — Erhebung zum Substantiv

z.B. punch — stanzen
puncher — der Stanzer
punching — das Stanzen

ed — als Kennzeichnung des Partizips Perfekt

z.B. check — Kontrollieren
checked — kontrolliert

Abkürzungsverzeichnis

zur Arbeit mit den Bürocomputern A 5120/30

Abkürzung	Herleitung	Bedeutung
AADR		Anfangsadresse
AAWA		Anfangsadresse des Anwenderbereiches (RAM)
AAWO		Anfangsadresse des Anwenderbereiches (ROM)
AB		Anfangsadresse des Bildschirmarbeitsspeichers
ACK		positive Antwort auf letzten gesendeten Block
ADR		Adresse
AGM		Anfangsadresse des Grundmodul
ALU	arithmetic logic unit	Arithmetisch-logische Einheit
AP		Abonnentenpunkt
BAB		Bildschirmarbeitsbereich
BC		Bedingungscode
BCR	byte counter read	Bytezähler bei Empfang
BL	block length	Blocklänge
BM		Bandmarke
BOE	beginning of extent	Bereichsanfang
BOT	beginning of tape	Bandanfang (Lochkennzeichnung)
BPF		Blockprüffolge
BSE	beginning save extent	Beginn des geschützten Bereiches
CTC	counter-timer-circuit	Zähler-Zeitgeber Schaltkreis
CPU	central prozessing unit	zentrale Verarbeitungseinheit
DA	device adress	logische Geräteadresse
DCB	data set control block	dynamischer Dateidefinitionsblock

Abkürzung	H e r l e i t u n g	Bedeutung
DDB	data set definition block	statischer Dateidefinitionsblock
DFÜ		Datenfernübertragung
DMA	direct memory access	direkter Speicherzugriff
DT	data set typ	Dateityp
EA; E/A		Eingabe-/Ausgabe-
ENQ		Abschluß einer Adressierungsfolge
EOD	end of data	Ende des belegten Datenbereiches (nächster freier Satz)
EOE	end of extend	Bereichsende
EOT	end of tape	• Bandende (Lochkennzeichnung) • Abbruch der Übertragung
ERAM		Ende des RAM-Bereiches
ESE	end of save extend	Ende des geschützten Bereiches
ETB		Blockendezeichen für Datenübertragung
ETR	end of track	Spurendekennsatz
ETX		Nachrichtenende
FF	form feed	neue Seite
INT		Interrupt
IOA	input-output area	Pufferbereich für den Datenblock (Adresse)
IOS	input-output size	Länge des Pufferbereiches
IT	index table	Indextabelle
ITC	index table creation	Erstellung Indextabelle
K.A.		keine Antwort
KL	key length	Schlüssellänge
KP	key positions	Schlüsselposition
LF	line feed	Zeilenschaltung
LIFO	last in-first out	zuletzt hinein - zuerst heraus (Arbeitsweise des Kellerspeichers)
LIT	length of index table	Länge der Indextabelle
LRC		Blockprüfzeichen

Abkürzung	Herleitung	Bedeutung
M		Monitor
MLR	message length read	maximale Nachrichtenlänge
MLW	write	(bei Empfang/Senden)
NAK		negative Antwort auf letzten gesendeten Block
NDA	number of device adresses	Anzahl der logischen Geräteadressen
NDR		nächster freier Datensektor
NFND	no found	nicht gefunden
NL	new line	neue Zeile
NT	number of tracks	Anzahl der durchsuchenden Spuren
OI	open indicator	Eröffnungsanzeiger
pa		physische Geräteadresse
PBL	physical block length	physische Blocklänge
PIO	parallel input-output	Parallel Eingabe-Ausgabe-Baustein
PRG		Programm
PSB		Programmstatusbit
PT	procedure typ	Prozedurtyp
RA	record adress	aktuelle Satzadresse
RAM	random access memory	Speicher für wahlfreien Zugriff (Lese-Schreib-Speicher)
RL	record length	Satzlänge
RLS		maximale Satzlänge
ROM	read only memory	nur Lese Speicher
R/W	read/write	Lesen/Schreiben
SEB	status error byte	Statusfehlerbyte
SFR/W	system flags read/write	Systemkennzeichen bei Empfang/Senden
SI	save indicator	Dateischutzindikator

Abkürzung	Herleitung	Bedeutung
SIO	seriell input-output	serieller Eingabe-Ausgabe-Baustein
SOH		Start des Kopfes einer Nachricht
STX		Blockanfang für Textübertragung (positive Antwort anstelle ACK)
SV		Symbolverzeichnis
SY		System
SYN		Synchronisation
TAB		Tabulation (geschützte Position)
TO	time out	Zeitunterbrechung
TP	tape	Band
UP		Unterprogramm
VSN	volume sequence number	Datenträgerfolgenummer
VT	vertical tabulation	Zeilenvorschub
WI	write indicator	Schreibindikator
ZRE		zentrale Recheneinheit
ZVE		zentrale Verarbeitungseinheit

Unget

Stand: 30.06.81

739128

Anwendungsbeschreibung der Dienst- und Hilfsprogramme für SIOS 1520

VEB Robotron-Buchungsmaschinenwerk
Karl-Marx-Stadt 1982

OK 0220.03

Die vorliegenden "Anwendungsbeschreibungen" entsprechen dem Stand vom 30.06.81. Vom Entwickler werden neben den in dieser Ausgabe enthaltenen weitere Programme geschaffen.

Nachdruck, jegliche Vervielfältigung dieser Unterlage oder Auszüge daraus sind unzulässig.

Die Auserbeitung dieser Dokumentation erfolgte durch ein Kollektiv der Abteilung "software-Entwicklung" des VEB Robotron-Buchungsmaschinenwerk Karl-Marx-Stadt.

Herausgeber:

VEB Robotron-Buchungsmaschinenwerk
Karl-Marx-Stadt
9010 Karl-Marx-Stadt
PSF 129

Verantw. Lektor und Gesamtbearbeitung
im Auftrag der DEWAG Dresden,
Zweigbetrieb Cottbus:

Dr. Lutz-Steffen Tag, Leipzig

A 2

Vorwort

Die vorliegenden "Anwendungsbeschreibungen" beinhalten Angaben zur Wirkungsweise und zu den gerätetechnischen Forderungen der Dienst- und Hilfsprogramme des SIOS 1520. Die Programme arbeiten nur unter Steuerung des SIEX/MINT.

Für weiterführende Studien werden folgende Schriften empfohlen:

- Anleitung für den Systemprogrammierer
- Sprachbeschreibung Assemblersprache MABS 1520
- Anleitung für den Programmierer der Dienst- und Hilfsprogramme
- Bedienungsanleitungen der Dienst- und Hilfsprogramme
- Bedienhandbücher der Geräte A 5120/A 5130

Kurzbez.	Inhalt
INI8	Initialisieren 8"-Diskette
INI5	Initialisieren 5,25"-Diskette
FGEN	Dateigenerierung Diskette
EDIT	Quellbucheditor
MBAS/CASS/SASS	Assembler
LINK	Linkage Editor
LBSV	Bibliotheksservice
MTES	Makroprogramm-Testsystem
TMIK	Testhilfe Mikrobefehlsfolgen
KONV	Konvertieren einer sequentiellen Datei in eine gestreute Datei
SCNG	Berechnung der Prüfziffer
DINH	Anzeige/Inhaltsdruck Sektorinhalt Diskette
DIKA	Anzeige Inhalt Datei/Kassette
DMPM	Druck/Anzeige RAM/ROM-Speicherinhalt
BKTR	Buchtransformation MEOS ⟶ SIOS
PHAK	Phase von Diskette nach Kassette
MIXD	Mischprogramm
COCA	Kopieren von Dateien von/nach Magnetbandkassette
CODP	Physisches Kopieren von Disketten
TEXT	Textverarbeitung
DIDI	Anzeige Inhalt Datei/Diskette
PRKA	Druck Inhalt Datei/Kassette
TFGC	Testdateien auf Magnetbandkassette
TFLI	Testdateien auf Diskette
PRDI	Druck Datei/Diskette

Kurzbez.	Inhalt
EM62	Emulation Abonnentenpunkt 62
COPD	Logisches Kopieren von Diskettendateien
SORD	Sortierprogramm Floppy Disk
CATM	Unitialisieren Kassette mit Bandmark / HDR
FUPR	Funktionsprüfprogramm

Anwendungsbeschreibung

Programmbezeichnung: Initialisieren von 8"-Disketten (256 K-Byte)

Kurzbezeichnung: INI8 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

Anlagen

1. Zweckbestimmung des Programms

1.1. Technische und anwendungstechnische Grundlagen

Die Datenspeicherung auf Disketten erfolgt für die "Geräte des Systems der Datenerfassung" auf der Grundlage folgender (inhaltlich weitgehend identischer) Standards:

- "Entwicklungsrichtlinie des Kombinats Robotron für die Datenträgerkompatibilität der 8"-Diskette" vom September 1979; KROS 5108
- ISO/TC 97/SC 11N 209 DP 5654
  "Aufzeichnungsformat für Standard-Diskette"
- Vornorm zur DIN 66238
  "Einseitig verwendbare flexible Magnetplatte 200 - beschriebene Magnetplatte"

- ECMA/TC 15/77/15
  "Flexible DISK Cartridge
  Labeling and File Structure for Information Interchange"
  vom Juli 1977
- IBM on-side-Diskette
  original equipment
  GA 21-91903

Für das Programm "Initialisieren von Disketten" sind folgende Festlegungen wesentlich (hierzu s. auch:"Systemhandbuch" Teil: Dateiarbeit):

- Es wird zwischen initialisierten und noch nicht bzw. nicht mehr initialisierten Disketten unterschieden.
  Initialisieren bedeutet:
  - Physisch Sektoren der Länge 128, 256, 512 oder 1024 Bytes wahlweise in einer vorzugebenden Sektorfolge (s. Anlage) in Form von 77 Spuren definiert auf die Diskette aufzubringen (Spur 00 immer in der Länge von 128 Bytes je Sektor);
  - den Datenträgerkennsatz in der Indexspur (Spur 00) definiert aufzubringen und den Bereich der Dateikennsätze vorzubereiten.
- Die möglichen Reihenfolgen der Sektoren bei den verschiedenen Sektorlängen sind in den Anlagen 1 - 4 dargestellt.
- Spur 00 muß immer fehlerfrei sein. Von den Spuren 1 ... 76 können bis zu 2 Spuren mit Fehlern sein. Diese werden beim Initialisieren gekennzeichnet und übergangen. Liegen keine fehlerhafte Spuren vor, so werden die beiden äußeren Spuren beim Initialisieren übergangen (Reserve).
  Nutzbar für den Anwender sind immer (77 - 2) = 75 Spuren, davon die Spur 00 als Indexspur.
- Für die Indexspur erfolgen alle Aufzeichnungen im DKOI-Code (nach ST RGW 358-76).

Es sind folgende Abweichungen zum EBCDIC-Code, der von verschiedenen Diskettenherstellern ebenfalls genutzt wird, zu beachten.

Codierung	DKOI	EBCDIC
4A	[	0
4F	!	!
5A	]	!
5F	^	¬
6A	/	^

1.2. Spezielle Zweckbestimmung

Mit dem Programm sollen
- fabrikneue Disketten,
- benutzte, Fehler aufweisende Disketten,
- benutzte, mit anderer Sektorlänge/Sektorfolge erwünschte Disketten,
- benutzte, inhaltlich "gelöscht" erwünschte Disketten

definiert neu beschrieben werden.
- benutzte Disketten (z.B. von anderen Geräten mit Daten gefüllt)

nur auf Spur Ø initialisiert werden.

Dabei soll es möglich sein:
- Sektorlänge und Sektorfolge (s. Anlagen) vorzugeben,
- fehlerhafte Spuren zu erkennen und zu behandeln (weniger als 3 Spuren, ≠ Spur 0) bzw. die Diskette für weitere Verarbeitungen zurückzuweisen (ab 3 defekten Spuren, Spur 0 defekt).

2.
Anwendungsbedingungen

- Im Laufwerk mit der Geräteadresse 05 ist die zu initialisierende Diskette einzulegen und zu verriegeln.
- Erfolgreiches Initialisieren ist nur möglich, wenn maximal 2 Spuren defekt sind, wobei Spur ØØ immer fehlerfrei sein muß.
- Erforderliche Geräteausstattung:
  - ≧ 1 Floppy-Disk-Laufwerk, logische Geräteadresse 05,
  - Bildschirm 1 oder 2 K Bytes,
  - Arbeitsspeicher ≧ 6,5 K Bytes.

3.
Beschreibung von Aufgabe und Lösung

3.1.
Aufgabe

Durch das Programm sind folgende Aufgaben zu lösen:
- Anfangskommunikation zur manuellen Eingabe von Sektorlänge und Sektorfolge, Initialisieren gesamt/nur Spur Ø,
- Prüfen der Arbeitsgeschwindigkeit des Laufwerkes,
- Beschreiben jeder Spur von Anfang bis Ende ohne Unterbrechung, Erkennen und Kennzeichnen fehlerhafter Spuren so, daß die kontinuierliche Numerierung der brauchbaren Spuren gewährleistet ist,
- Abbruch der Arbeit und Information an den Bediener, wenn zu viele Spuren defekt sind bzw. ein Gerätefehler vorliegt,
- Information an den Bediener bei erfolgreichem Abschluß der Initialisierung,
- Nebenbedingung ist, daß das Initialisieren möglichst rasch ablaufen soll.

3.2.
Lösung

Das Programm löst die gestellte Aufgabe, insbesondere die Bedingungen:

- Beschreiben jeder Spur von Anfang bis Ende ohne Unterbrechung,
- Möglichst rascher Programmablauf

dadurch, daß nach der (in Maschinensprache geschriebenen) Anfangskommunikation das gesamte Programm in der CPU-Sprache (Mikroprogrammierung) geschrieben ist.

Dabei wird das Betriebssystem (SIEX) teilweise außer Betrieb gesetzt. Die Moduln für den Speicherschutz und gewisse Zeitunterbrechungen sind abgeschaltet. Die Folgen davon sind:

- Fehler im Programmablauf werden nur durch das Initialisierungsprogramm erkannt.
  Sie werden angezeigt (Anzeige durch das Initialisierungsprogramm lt. Bedienungsanweisung). Eine Behandlung der Fehler durch das Anwendungsprogramm erfolgt nicht.

In beiden Fällen muß die Anlage durch RESET, CI-Taste oder Ausschalten/Einschalten in den Grundzustand versetzt werden. Eine Programmfortsetzung über den "Monitor" durch erneuten Start ist möglich.

- Bei korrektem Ablauf des Programms bis zum Ende verharrt das Programm nach der entsprechenden Anzeige auf dem Bildschirm in einer Warteschleife. Die initialisierte Diskette ist zu entnehmen. Die Anlage muß durch RESET, CI-Taste oder Ausschalten/Einschalten in den Grundzustand versetzt werden.

Das Initialisieren einer Diskettenseite dauert, wenn keine Wiederholungen im Ablauf erforderlich sind, etwa 3 Minuten.

Das Programm läuft nach Übernahme der eingegebenen Größen wie folgt ab:

- Modifizieren des Betriebssystems,
- Kontrolle auf Vorhandensein einer Diskette (Laufwerk 5), der Eingaben und der Diskettengeschwindigkeit,
- Aufzeichnung ab Spur ØØ bis Spur 76 fortlaufend ohne Unterbrechung von Indexloch zu Indexloch. Dabei arbeiten beide CPU zeitparallel (1. CPU-Steuerung der Aufzeichnung, 2. CPU-Vorbereitung des Inhalts des nächsten Sektors).
- Kontrollesen von Spur 76 bis Spur ØØ.
- Fehlerhafte Spuren werden mit einem speziellen Code initialisiert, für sie erfolgt kein Kontrollesen.
- Nach dem Kontrollesen der 76. Spur wird (zeitparallel zur Bewegung des Kopfes nach außen) kontrolliert, ob alle Spuren aufgezeichnet sind. Werden Fehler festgestellt, so wird das Initialisieren noch maximal zweimal wiederholt. Bei weiteren Fehlern erfolgt die Meldung: Gerätefehler.
- Nach erfolgreichem Abschluß des Prüfens wird die Spur ØØ noch einmal initialisiert, um die Nummer defekter Spuren einzutragen.
- Auf dem Bildschirm wird die Schlußmeldung angezeigt.

4. Ein- und Ausgangsdaten

Eingangsdaten

- Zu initialisierende Diskette,
- Sektorlänge (128, 256, 512, 1024 Bytes),
- Sektorfolge (Ø1 -yy, yy je nach Sektorlänge),
- Nummern (xx) bekannter defekter Spuren.

Ausgangsdaten

- Initialisierte Diskette (bzw. nur neu initialisierte Spur Ø)und Schlußmeldung

oder

- Fehlermeldung (vom Initialisierungsprogramm) und Diskette nicht definierten Inhalts.

Abkürzungsverzeichnis

CPU	Zentrale Steuereinheit
EBCDIC-Code	extended binary coded decimal interchange code
ECMA	(West-) Europäische Vereinigung der Computerhersteller
ISO-Norm	Internationale Organisation für Standardisierungs-Norm
SIEX	Ausführungsteil des Betriebssystems SIOS
SIOS	Betriebssystem auf Basis der Zentraleinheit K 2526
ST-RGW	Standard/Norm im RGW

Anlage

Physische Satzfolge der Sektoren

Reihenfolge bei 128 Byte physischer Satzlänge (Sektorlänge)

Sektoren	Variationen												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	3	4	5	6	7	8	9	10	11	12	13	14
3	3	5	7	9	11	13	15	17	19	21	23	25	2
4	4	7	10	13	16	19	22	25	2	2	2	2	15
5	5	9	13	17	21	25	2	2	11	12	13	14	3
6	6	11	16	21	26	2	9	10	20	22	24	26	16
7	7	13	19	25	2	8	16	18	3	3	3	3	4
8	8	15	22	2	7	14	23	26	12	13	14	15	17
9	9	17	25	6	12	20	3	3	21	23	25	4	5
10	10	19	2	10	17	26	10	11	4	4	4	16	18
11	11	21	5	14	22	3	17	19	13	14	15	5	6
12	12	23	8	18	3	9	24	4	22	24	26	17	19
13	13	25	11	22	8	15	4	12	5	5	5	6	7
14	14	2	14	26	13	21	11	20	14	15	16	18	20
15	15	4	17	3	18	4	18	5	23	25	6	7	8
16	16	6	20	7	23	10	25	13	6	66	17	19	21
17	17	8	23	11	4	16	5	21	15	16	7	8	9
18	18	10	26	15	9	22	12	6	24	26	18	20	22
19	19	12	3	19	14	5	19	14	7	7	8	9	10
20	20	14	6	23	19	11	26	22	16	17	19	21	23
21	21	16	9	4	24	17	6	7	25	8	9	10	11
22	22	18	12	8	5	23	13	15	8	18	20	22	24
23	23	20	15	12	10	6	20	23	17	9	10	11	12
24	24	22	18	16	15	12	7	8	26	19	21	23	25
25	25	24	21	20	20	18	14	16	9	10	11	12	13
26	26	26	24	24	25	24	21	24	18	20	22	24	26

Anlage

Physische Satzfolge der Sektoren

Reihenfolge bei 256 Byte physischer Satzlänge (Sektorlänge)

Sektoren	Variationen							
	01	02	03	04	05	06	07	08
1	1	1	1	1	1	1	1	1
2	2	3	4	5	6	7	8	9
3	3	5	7	9	11	13	15	2
4	4	7	10	13	2	2	2	10
5	5	9	13	2	7	8	9	3
6	6	11	2	6	12	14	3	11
7	7	13	5	10	3	3	10	4
8	8	15	8	14	8	9	4	12
9	9	2	11	3	13	15	11	5
10	10	4	14	7	4	4	5	13
11	11	6	3	11	9	10	12	6
12	12	8	6	15	14	5	6	14
13	13	10	9	4	5	11	13	7
14	14	12	12	8	10	6	7	15
15	15	14	15	12	15	12	14	8

Anlage

Physische Satzfolge der Sektoren

Reihenfolge bei 512 Byte physischer Satzlänge (Sektorlänge)

Sektoren	Variationen 01	02	03	04
1	1	1	1	1
2	2	3	4	5
3	3	5	7	2
4	4	7	2	6
5	5	2	5	3
6	6	4	8	7
7	7	6	3	4
8	8	8	6	8

Physische Satzfolge der Sektoren

Reihenfolge bei 1024 Byte physischer Satzlänge (Sektorlänge)

Sektoren	Variationen 01	02
1	1	1
2	2	3
3	3	2
4	4	4

Anwendungsbeschreibung

Programmbezeichnung: Initialisieren von 5,25"-Disketten
Kurzbezeichnung: INI5 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

Anlagen

1. Zweckbestimmung des Programms

1.1. Technische und anwendungstechnische Grundlagen

Die Datenspeicherung auf 5,25"-Disketten erfolgt für die "Bürocomputer" auf der Grundlage folgender Standards:

- "Entwurf der Entwicklungsrichtlinie des Kombinates Robotron für die Datenträgerkompatibilität der 5,25"-Diskette vom April 1981; KROS 5110/01."
- ISO/TC 97/ SC 11N 209 bzw. 347 "Aufzeichnungsformat für Standard-Diskette"
- Vornorm zur DIN 66247, Teil I.

Für das Programm "Initialisieren von Disketten" sind folgende Festlegungen wesentlich; hierzu s. auch:

"Systemhandbuch" Teil (Dateiarbeit):

- Man unterscheidet zwischen initialisierten und noch nicht bzw. nicht mehr initialisierten Disketten.
  Initialisieren bedeutet:
  . physisch Sektoren der Länge 128, 256, 512 oder 1024 Bytes wahlweise in einer vorzugebenden Sektorfolge (s. Anlage) in Form von 39 Spuren auf die Diskette definiert aufzubringen (Spur ØØ immer in der Länge von 128 Bytes je Sektor),
  . in der Indexspur (Spur ØØ den Datenträgerkennsatz definiert aufzubringen und den Bereich der Dateikennsätze vorzubereiten.
- Die möglichen Reihenfolgen der Sektoren bei den verschiedenen Sektorlängen sind in den Anlage 1 - 4 dargestellt.
- Spur ØØ muß immer fehlerfrei sein, von den Spuren 1 ... 39 können bis zu 2 Spuren fehlerhaft sein - diese werden beim Initialisieren gekennzeichnet und übergangen. Liegen keine fehlerhaften Spuren vor, so werden die beiden äußeren Spuren beim Initialisieren übergangen (Reserve).
  Für den Anwender sind immer (39-2) = 37 Spuren nutzbar, davon die Spur ØØ als Indexspur.
- Für die Indexspur erfolgen alle Aufzeichnungen im DKOI-Code (nach ST RGW 358-76).
  Folgende Abweichungen zum EBCDIC-Code, der von verschiedenen Diskettenherstellern ebenfalls genutzt wird, sind zu beachten:

Codierung	DKOI	EBCDIC
4A	[	0
4F	!	!
5A	]	!
5F	^	¬
6A	/	^

1.2. Spezielle Zweckbestimmung

Mit dem Programm sollen
- fabrikneue Disketten,
- benutzte, fehler aufweisende Disketten,
- benutzte, mit anderen Sektorlänge/Sektorfolge erwünschte Diskette,
- benutzte, inhaltlich "gelöscht" gewünschte Disketten definiert neu beschrieben werden,
- benutzte Disketten (z.B. von anderen Geräten mit Daten gefüllt) nur auf Spur Ø initialisiert werden.

Dabei sollen
- Sektorlänge und Sektorfolge (s. Anlagen) vorgebbar sein,
- fehlerhafte Spuren erkannt und behandelt werden (weniger als 3 Spuren, ≠ Spur 0) bzw. die Diskette für weitere Verarbeitungen zurückgewiesen werden (ab 3 defekte Spuren, Spur 0 defekt).

2. Anwendungsbedingungen

- Im Laufwerk mit der Geräteadresse Ø5 ist die zu initialisierende Diskette einzulegen und zu verriegeln.
- Erfolgreiches Initialisieren ist nur möglich, wenn maximal 2 Spuren defekt sind, wobei Spur ØØ immer fehlerfrei sein muß.
- Erforderliche Gerätesausstattung:
  . ≥ 1 Floppy-Disk-Laufwerk, logische Geräteadresse Ø5,
  . Bildschirm 1 oder 2 K Bytes,
  . Arbeitsspeicher ≥ 6,5 K Bytes.

3.
Beschreibung von Aufgaben und Lösung

3.1.
Aufgabe

Durch das Programm sind folgende Aufgaben zu lösen:
- Anfangskommunikation zur manuellen Eingabe von Sektorlänge und Sektorfolge, Initialisieren gesamt/nur Spur Ø.
- Prüfen der Arbeitsgeschwindigkeit des Laufwerkes,
- Beschreiben jeder Spur von Anfang bis Ende ohne Unterbrechung, Erkennen und Kennzeichnen fehlerhafter Spuren so, daß die kontinuierliche Numerierung der brauchbaren Spuren gewährleistet ist,
- Abbruch der Arbeit und Information an den Bediener, wenn zu viele Spuren defekt sind bzw. ein Gerätefehler vorliegt,
- Information an den Bediener bei erfolgreichem Abschluß der Initialisierung,
- Nebenbedingung ist, daß das Initialisieren möglichst rasch ablaufen soll.

3.2.
Lösung

Das Programm löst die gestellte Aufgabe, insbesondere die Bedingungen:
- Beschreiben jeder Spur von Anfang bis Ende ohne Unterbrechung,
- Möglichst rascher Programmablauf

dadurch, daß nach der in Maschinensprache geschriebenen Anfangskommunikation das gesamte Programm in der CPU-Sprache (Mikroprogrammierung) geschrieben ist.

Dabei wird das Betriebssystem (SIEX) teilweise außer Funktion gesetzt. Die Moduln für den Speicherschutz und gewisse Zeitunterbrechungen sind abgeschaltet. Die Folgen davon sind:

- Fehler im Programmablauf werden durch das Initialisierungsprogramm erkannt.
- Sie werden angezeigt (Anzeige durch das Initialisierungsprogramm lt. Bedienungsanweisung).
  Die Fehler werden durch das Anwendungsprogramm nicht behandelt.

In beiden Fällen muß die Anlage durch RESET, CI-Taste oder Ausschalten/Einschalten in den Grundzustand versetzt werden. Eine Programmfortsetzung über den "Monitor" durch erneuten Start ist möglich.

- Bei korrektem Ablauf des Programms bis zum Ende verharrt das Programm nach der entsprechenden Anzeige auf dem Bildschirm in einer Warteschleife. Die initialisierte Diskette ist zu entnehmen. Die Anlage muß durch RESET, CI-Taste oder Ausschalten/Einschalten in den Grundzustand versetzt werden!

Das Initialisieren einer Diskettenseite dauert ungefähr 2 Min., wenn keine Wiederholungen im Ablauf erforderlich sind.

Das Programm läuft nach Übernahme der eingegebenen Größen wie folgt ab:

- Modifizieren des Betriebssystems,
- Kontrolle auf Vorhandensein einer Diskette (Laufwerk 5), der Eingaben und der Diskettengeschwindigkeit,
- Aufzeichnung ab Spur ØØ bis Spur 39 fortlaufend ohne Unterbrechung von Indexloch zu Indexloch. Dabei arbeiten beide CPU zeitparallel (1. CPU-Steuerung der Aufzeichnung, 2. CPU-Vorbereitung des Inhalts des nächsten Sektors).
- Kontrollesen von Spur 39 bis Spur ØØ.

- Fehlerhafte Spuren werden mit einem speziellen Code initialisiert, für sie erfolgt kein Kontrollesen.
- Nach dem Kontrollesen der 39. Spur wird (zeitparallel zur Bewegung des Kopfes nach außen) kontrolliert, ob alle Spuren aufgezeichnet sind. Werden Fehler festgestellt, so wird das Initialisieren noch maximal zweimal wiederholt. Bei weiteren Fehlern erfolgt die Meldung: Gerätefehler.
- Nach erfolgreichem Abschluß des Prüfens wird die Spur ØØ noch einmal initialisiert, um die Nummer defekter Spuren einzutragen.
- Auf Bildschirm wird die Schlußmeldung angezeigt.

4. Ein- und Ausgangsdaten

Eingangsdaten

- Zu initialisierende Diskette.
- Sektorlänge (128, 256, 512, 1024 Bytes).
- Sektorfolge (Ø1-YY), YY je nach Sektorlänge,
- Nummern (xx) bekannter defekter Spuren.

Ausgangsdaten

- Initialisierende Diskette (bzw. nur neu initialisierte Spur Ø) und Schlußmeldung oder
- Fehlermeldung vom Initialisierungsprogramm und Diskette nicht definierten Inhalts.

Abkürzungsverzeichnis

CPU	Zentrale Steuereinheit
EBCDIC-Code	extended binary coded decimal interchange code
ISO-Norm	Internationale Organisation für Standardisierungs-Norm
SIEX	Ausführungsteil des Betriebssystems SIOS
SIOS	Betriebssystem auf der Basis der Zentraleinheit K 2526
ST-RGW	Standard/Norm im RGW

Anlage

Physische Satzfolge der Sektoren

Reihenfolge bei 128 Byte physischer Satzlänge (Sektorlänge)

Sektoren	Variationen												
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	3	4	5	6	7	8	9	10	11	12	13	14
3	3	5	7	9	11	13	15	17	19	21	23	25	2
4	4	7	10	13	16	19	22	25	2	2	2	2	15
5	5	9	13	17	21	25	2	2	11	12	13	14	3
6	6	11	16	21	26	2	9	10	20	22	24	26	16
7	7	13	19	25	2	8	16	18	3	3	3	3	4
8	8	15	22	2	7	14	23	26	12	13	14	15	17
9	9	17	25	6	12	29	3	3	21	23	25	4	5
10	10	19	2	10	17	26	10	11	4	4	4	16	18
11	11	21	5	14	22	3	17	19	13	14	15	5	6
12	12	23	8	18	3	9	24	4	22	24	26	17	19
13	13	25	11	22	8	15	4	12	5	5	5	6	7
14	14	2	14	26	13	21	11	20	14	15	16	18	20
15	15	4	17	3	18	4	18	5	23	25	6	7	8
16	16	6	20	7	23	10	25	13	6	66	17	19	21
17	17	8	23	11	4	16	5	21	15	16	7	8	9
18	18	10	26	15	9	22	12	6	24	26	18	20	22
19	19	12	3	19	14	5	19	14	7	7	8	9	10
20	20	14	6	23	19	11	26	22	16	17	19	21	23
21	21	16	9	4	24	17	6	7	25	8	9	10	11
22	22	18	12	8	5	23	13	15	8	19	20	22	24
23	23	20	15	12	10	6	20	23	17	9	10	11	12
24	24	22	18	16	15	12	7	8	26	19	21	23	25
25	25	24	21	20	20	18	14	16	9	10	11	12	13
26	26	26	24	24	25	24	21	24	18	20	22	24	26

Anlage

Physische Satzfolge der Sektoren

Reihenfolge bei 256 Byte physischer Satzlänge (Sektorlänge)

Sektoren	Variationen							
	01	02	03	04	05	06	07	08
1	1	1	1	1	1	1	1	1
2	2	3	4	5	6	7	8	9
3	3	5	7	9	11	13	15	2
4	4	7	10	13	2	2	2	10
5	5	9	13	2	7	8	9	3
6	6	11	2	6	12	14	3	11
7	7	13	5	10	3	3	10	4
8	8	15	8	14	8	9	4	12
9	9	2	11	3	13	15	11	5
10	10	4	14	7	4	4	5	13
11	11	6	3	11	9	10	12	6
12	12	8	6	15	14	5	6	14
13	13	10	9	4	5	11	13	7
14	14	12	12	8	10	6	7	15
15	15	14	15	12	15	12	14	8

Anlage

Physische Satzfolge der Sektoren

Reihenfolge bei 512 Byte physischer Satzlänge (Sektorlänge)

	Variationen			
Sektoren	01	02	03	04
1	1	1	1	1
2	2	3	4	5
3	3	5	7	2
4	4	7	2	6
5	5	2	5	3
6	6	4	8	7
7	7	6	3	4
8	8	8	6	8

Physische Satzfolge der Sektoren

Reihenfolge bei 1024 Byte physischer Satzlänge (Sektorlänge)

	Variationen	
Sektoren	01	02
1	1	1
2	2	3
3	3	2
4	4	4

Anwendungsbeschreibung

Programmbezeichnung: Generierung von Kennsätzen auf 8"- bzw. 5"-Disketten
Kurzebezeichnung: FGEN 152Ø (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Der Anwender soll imstande sein, selbst Dateien aufzubauen und zu pflegen. Als Datenträger dient die Diskette.

Für die Arbeit mit den Dateibefehlen müssen folgende Voraussetzungen erfüllt sein:

- Auf Sektor Ø7 Spur ØØ muß sich der Datenträgerkennsatz (VOL1) befinden.
- Ab Sektor Ø8 Spur ØØ müssen die Dateikennsätze (HDR1 bzw. DDR1 für gelöschte Dateikennsätze) stehen. Die Spur ØØ ist für die Speicherung der Kennsätze reserviert. Mit dem Hilfsprogramm "FGEN" (FILE GENERATION) können diese Voraussetzungen geschaffen werden.

2. Anwendungsbedingungen

Es sind Disketten zu verwenden, die durch das Programm "INIT8" bzw. für Minidisketten "INIT5" initialisiert worden

sind bzw. deren physischer Aufbau dem im Programm INIT bzw. dem im Systemhandbuch, Teil: Anleitung für den (System-) Programmierer, Teil III, genannten Standard entspricht.

Technische Mittel:

- Arbeitsspeicher 13 K Byte
- Externe Geräte
  - Bildschirm (1Ø24 oder 192Ø)
  - Floppy-Laufwerk
  - Drucker

3. Beschreibung von Aufgabe und Lösung

3.1. Aufgabe

Mit Hilfe von FGEN sollen folgende Funktionen realisiert werden:

- Auf Disketten ohne Datenträgerkennsatz soll ein neuer aufgebracht werden.
- Ändern eines vorhandenen Datenträgerkennsatzes,
- Anzeige bzw. Druck des Datenträgerkennsatzes und eines durch Namen ausgewählten Dateikennsatzes,
- Neueingabe, Änderung, Löschen eines Dateikennsatzes,
- Anzeige bzw. Druck des Inhaltsverzeichnisses und der freien Datenbereiche einer Diskette.

Mit Hilfe dieser Funktionen erfolgt die physische Speicherplatzzuweisung für logische Dateien!

3.2.
Lösung

Mit Hilfe des Programms FGEN können 8" Standarddisketten und 5" Minidisketten "verwaltet" werden. Nach dem Laden des Programms erfolgt die Menü-Anzeige. Über Steuerziffern werden die logischen Geräte zugewiesen, und über Buchstaben kann das gewünschte Programmteil ausgewählt werden.

Es sind folgende Zuweisungen möglich:

1 = Zuweisung Drucker 1
2 = Zuweisung Drucker 2
3 = Zuweisung Floppy-Laufwerk
4 = Unterbrechung aller Verbindungen und Halt im Monitorstatus

Es können folgende Teilfunktionen ausgewählt werden:

A = Neueingabe Datenträgerkennsatz,
B = Ändern Datenträgerkennsatz,
C = Anzeige Datenträger- und eines Dateikennsatzes,
D = Druck Datenträger- und eines Dateikennsatzes,
E = Neueingabe eines Dateikennsatzes,
F = Ändern eines Dateikennsatzes,
G = Löschen eines Dateikennsatzes,
H = Anzeige des Inhaltsverzeichnisses und der freien Bereiche,
I = Druck des Inhaltsverzeichnisses und der freien Bereiche.

Zu Steuerziffer 1

Es wird das logische Gerät ØB zugewiesen, und es muß die Anzahl Zeilen je Seite eingegeben werden.
Es erfolgt eine Prüfung der Zuweisung. Bei Fehler erfolgt eine Fehlermeldung.

Zu Steuerziffer 2

Es wird das logische Gerät ØC zugewiesen, und es muß die Anzahl Zeilen je Seite eingegeben werden.
Es erfolgt eine Prüfung der Zuweisung. Bei Fehler erfolgt eine Fehlermeldung.

Zu Steuerziffer 3

Es muß die logische Geräte-Nr. des gewünschten Floppy-Laufwerkes eingegeben werden.
Es erfolgt eine Prüfung auf Zulässigkeit der Geräte-Nr.

Zu Steuerziffer 4

Es werden alle Verbindungen des Programms aufgehoben.
Das Programm stoppt im Monitorstatus.

Zu Programmteil A

Es erfolgt eine Kontrolle, ob dieses Programmteil zulässig ist oder nicht. Ist schon ein Datenträgerkennsatz vorhanden, so wird dieses Programmteil mit einer Fehlermeldung abgewiesen. Ansonsten wird ein Standarddatenträgerkennsatz eingestellt und zur Änderung freigegeben.
Nach Quittierung für die Richtigkeit der Eingabe erfolgt eine formelle und logische Prüfung. Die Angaben zum physischen Aufbau der Diskette werden nicht überprüft. Werden Fehler erkannt, so erfolgt eine Fehlermeldung, und die falsche Position wird unterstrichen. Danach kann korrigiert werden.

Zu Programmteil B

Es wird der vorhandene Datenträgerkennsatz eingelesen und angezeigt.

Ist kein Datenträgerkennsatz auf der Diskette, so wird dieses Programmteil mit einer Fehlermeldung abgewiesen.

Es können alle logischen Positionen geändert werden. Nach Quittierung für die Richtigkeit der Eingaben erfolgt eine formelle Prüfung. Werden Fehler erkannt, so erfolgt eine Fehlermeldung, und die falsche Position wird unterstrichen. Danach kann korrigiert werden.

Zu Programmteil C

Es wird der Datenträgerkennsatz eingelesen und angezeigt. Danach muß der Dateiname des gewünschten Dateikennsatzes eingegeben werden. Dieser wird auf der Diskette gesucht. Wird der Name gefunden, so erfolgt die Anzeige des Dateikennsatzes, und es kann der nächste Name eingegeben werden. Wird der Name nicht auf der Diskette gefunden, so erfolgt eine Fehlermeldung. Ein neuer Dateiname kann eingegeben werden.

Zu Programmteil D

Es wird der Datenträgerkennsatz eingelesen und gedruckt. Danach muß der Dateiname des gewünschten Dateikennsatzes eingegeben werden. Dieser wird auf der Diskette gesucht. Ist der Name gefunden, so erfolgt der Druck dieses Dateikennsatzes, und es kann ein neuer Dateiname eingegeben werden.

Wird der Name nicht gefunden, so erfolgt eine Fehlermeldung, und es kann ein neuer Dateiname eingegeben werden.

Zu Programmteil E

Es erfolgt eine Prüfung, ob die Möglichkeit einer Neueingabe eines Dateikennsatzes besteht. Im negativen Fall erfolgt eine Fehlermeldung.

Liegt die Möglichkeit der Neueingabe vor, so werden alle Dateikennsätze gelesen und gespeichert. Danach erfolgt die Anzeige einer Standerdvorgabe, die geändert werden muß.

Nach Quittierung für die Richtigkeit der Eingabe erfolgt eine formelle und logische Prüfung. Werden Fehler erkannt, so erfolgt eine Fehlermeldung. Die fehlerhafte Position wird unterstrichen.
Danach muß korrigiert werden.

Zu Programmteil F

Es muß der Dateiname des gewünschten Dateikennsatzes eingegeben werden. Dieser wird auf der Diskette gesucht. Wird er nicht gefunden, so erfolgt eine Fehlermeldung. Die Eingabe kann dann wiederholt werden.
Ist der Dateikennsatz gefunden, so wird dieser angezeigt und zur Änderung freigegeben.
Nach Quittierung für die Richtigkeit der Eingabe erfolgt eine formelle und logische Prüfung. Werden Fehler erkannt, so erfolgt eine Fehlermeldung, und die fehlerhafte Position wird unterstrichen.
Danach muß korrigiert werden.

Zu Programmteil G

Es muß der Dateiname des zu löschenden Dateikennsatzes eingegeben werden. Dieser wird auf der Diskette gesucht. Wird er nicht gefunden, so erfolgt eine Fehlermeldung, und die Eingabe muß wiederholt werden.
Ist der Dateiname gefunden, so wird dieser als gelöscht gekennzeichnet angezeigt.
Danach muß die Richtigkeit quittiert werden.

Zu Programmteil H

Es wird der Datenträgerkennsatz eingelesen und angezeigt.
Nach Quittierung für Fortsetzung werden alle Dateikennsätze eingelesen und angezeigt.
Danach werden die freien Bereiche auf der Diskette ermittelt und angezeigt.

Zu Programmteil I

Es wird der Datenträgerkennsatz eingelesen und gedruckt, danach die Dateikennsätze, die freien Bereiche ermittelt und gedruckt.

4. Eingangs- und Ausgangsdaten

4.1. Eingangsdaten

- Initialisierte oder "benutzte" Diskette,
- Bedienereingaben.

4.2. Ausgangsdaten

- Diskette mit ggf. geänderten/neu aufgebrachten Datenträger- und/oder Dateikennsätzen,
- Listen.

Anwendungsbeschreibung

Programmbezeichnung: Programm zur Eingabe und Pflege von Quellprogrammen
Kurzbezeichnung: EDIT 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgaben und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

1.1. Anwendungstechnische Grundlagen

Die grundlegenden Zusammenhänge zwischen den von SIOS 1520 genutzten drei Bibliotheken und jenen Dienstprogrammen, die diese Bibliotheken füllen bzw. ihren Inhalt verarbeiten, ist im Systemhandbuch, Teil Programmspeicherung, Programmerzeugung (Einleitung), beschrieben.
Die Aufgaben und der Aufbau der Quelltextbibliothek sind unter Pkt. 1.5.1. beschrieben.

1.2. Spezielle Zweckbestimmung

Das Dienstprogramm EDITOR ermöglicht es, die verschiedenen Quelltexte dialogorientiert in die Quelltextbibliothek (source library-SL) einzugeben, zu ändern, zu erweitern, ganz oder teilweise zu löschen, anzuzeigen oder zu drucken.

2.
Anwendungsbedingungen

- Als Datendisketten sind nur Bibliotheksdisketten nach SIOS 1520 zugelassen.
- Für die Pflegefunktionen auf diesen Funktionen ist das Programm Bibliotheksservice zu nutzen.

Technische Mittel

- Tastatur mit lateinischen Groß- und Kleinbuchstaben, mindestens jedoch lat. Großbuchstaben,
- Bildschirm 1024 oder 1920 Zeichen
  (bei Bildschirm mit 1920 Zeichen werden 1024 Zeichen mittig genutzt),
- Arbeitsspeicher $\geq$ 19 K Byte
- Wahlweise Drucker mit Leporello oder Journal
- $\geq$ 1 Laufwerk Floppy Disk

3.
Beschreibung von Aufgabe und Lösung

3.1.
Beschreibung der Aufgabe

Durch das Programm sind folgende Aufgaben zu lösen:
- Realisieren eines speziellen Kommunikations- und Kommandosystems, um die Funktionen bedienerfreundlich aufrufen und verlassen zu können,
- (Erst-) Eingabe von Quellbüchern bzw. ihrer Teile; dabei wahlweise simultaner Druck,
- Einfügen/Ändern von Quellbüchern,
- Mischen von Quellbuchteilen,
- Löschen von Quellbuchteilen,

- Anzeige von Teilen bzw. gesamten Büchern,
- Druck von Teilen bzw. gesamten Büchern,
- Anzeige von Quellbibliotheksverzeichnissen,
- Verwaltung der Quellbibliothek während ihrer Nutzung.

Wichtige Randbedingungen sind:

- Zeitgünstige und bedienerfreundliche Abläufe speziell bei der (Erst-) Eingabe von Quellbüchern durch Simultanarbeit,
- Syntaxkontrolle (mit ausgewählten Funktionen) für Makroquellzeilen,
- Wahlweise Nutzung des Druckers,
- RAM-Arbeitsspeicherbedarf $\geq$ 18 K Byte,
- Wahlweise Arbeit mit Bildschirm 1 K oder 2 K.

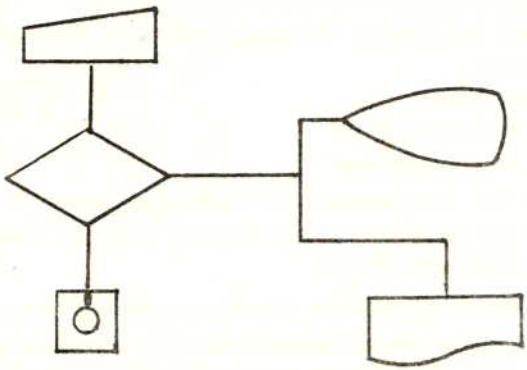
3.2. Beschreibung der Lösung

Das Programm löst die gestellten Aufgaben- die gestellten Randbedingungen erfüllend - dadurch, daß

- eine modulare Programmstruktur geschaffen wurde,
- eine einheitliche Kommandostruktur-Nutzung der Starttasten realisiert wurde,
- ausgewählte Programmteile bzw. Unterprogramme in CPU-Sprache geschrieben sind.

Alle Teilfunktionen des Editors stellen abgeschlossene Programmteile dar, die von einem zentralen Punkt (Kommandoauswahl) aus gestartet werden können.
Im Programm werden Eingaben/Startkommandos geprüft, fehlerhafte Eingaben zurückgewiesen.
Die allgemeine Fehlerbehandlung der E/A-Geräte wird durch den EDITOR selbst realisiert.

3.2.1.
Übersicht: Kommunikationsmittel



3.2.2.
Tastatur/Bedienerführung

Die Bedienung des Editors erfolgt über die alphanumerische Tastatur. Die Bedienerführung erfolgt über Bildschirm.
Die Textendetasten werden in allen Programmteilen allgemein einheitlich eingesetzt:

- Abschluß einer Eingabe ET1
- Rücksprung an vorherige Ebene oder an zentralen Punkt ET2

Ausnahme:

Im Programmteil Bucheingabe (Kommando 1 und 2) wird die Taste S1 zur Eingabewiederholung ("Zurückholen" einer Eingabe) genutzt.

3.2.3. Bildschirm

Die Bildschirmanzeige erfüllt zwei Funktionen:

1. Anzeige aus dem Inhalt der Quelltextbibliothek und Kommandoangebote
2. Dialogfunktionen:
   - Anzeige von Kommandos
   - Anzeige von Buchnamen
   - Anzeige von Anweisungen zur Bedienerführung
   - Anzeige von Parametern zur Abarbeitung des Kommandos
   - Anzeige Statusmeldungen

Dieser Bildschirmbereich wird als Editor-Dialogbereich definiert. Deshalb wurde der Bildschirmbereich im Verhältnis 10:6 geteilt, d.h. die oberen 10 Bildschirmzeilen dienen der Funktion 1, die unteren 6 Bildschirmzeilen der Funktion 2.

Der untere Bereich erhielt folgende Zuordnung:
(16-Zeilen-Bildschirm)

11. BS-Zeile - Anzeige Statusmeldungen
12. BS-Zeile - Anzeige Bedienanweisungen
13. BS-Zeile - Parameterzeile zur Spezifizierung der Kommandos
14. BS-Zeile - Parameterzeile für Zielbuch/Kommando

3.2.4. Drucker

Der Drucker liefert dem Programmierer wahlweise ein Erfassungsprotokoll. Dabei wird zu jeder Quellzeile die hexadezimale Quellbibliotheksadresse gedruckt. Es ist ebenfalls möglich, den Bibliotheksinhalt ganz oder auszugsweise unabhängig von einer Erfassung oder Korrektur zu drucken.

4.
Ein- und Ausgabedaten

4.1.
Eingabedaten

- Kommandos
  Sie dienen der Programmsteuerung durch den Operator.
  Typisch ist die Dialogarbeit: Operator - Gerät über Tastatur und Bildschirm.
- Zeilen eines Quellbuches
  Von Diskette bzw. Teile eines Quellbuches oder ein gesamtes Quellbuch
- Eingaben über Tastatur
  In Form von Zeilen (Teilen von Zeilen) eines Quellbuches, mit dem Ziel der/des
  - Ersteingabe
  - Änderung
  - Erweiterung
  - Löschens.

4.2.
Ausgabedaten

- Anzeigen des Programms auf dem Bildschirm:
  - Eingabedaten
  - Aufforderungen des Programmes (Marken)
  - Fehlermeldungen vom Programm/Betriebssystem.
- Druck von
  - Eingabedaten
  - Dateien
    bzw. Teilen von ihnen!

- Zeilen/Teile eines Quellbuches bzw. ein gesamtes Quellbuch auf Diskette
- Löschen von Buchteilen auf Diskette.

Anwendungsbeschreibung

Programmbezeichnung: Makrobefehlsassembler/CPU-Befehls-
assembler/SIOS-System-Assembler
Kurzbezeichnung: MBAS/CASS/SASS 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1.
Zweckbestimmung

Die Dienstprogramme MBAS 1520 (SIEX) (MBAS Makro-Befehls-Assembler), CASS 1520 (SIEX) (CASS CPU-Befehls-ASSembler) und SASS 1520 (SASS SIOS-System-ASSembler) sind Bestandteil des Servicekomplexes EDITOR-ASSEMBLER-BINDER-BIBLIOTHEKS-SERVICE. Sie dienen dem Anwender zur effektiven Erarbeitung eigener Programme.

Effektive Erarbeitung bedeutet:

- Kurze Programmerarbeitungszeit,
- Reduzierung der Test- und Änderungszeit,
- Verschiebbarkeit der Programme (Phasen),
- Verknüpfbarkeit der Teilprogramme (MODULE), Einfügbarkeit (vorgefertigter) Programme,
- Fehlerkontrolle und Anzeige von formalen Fehlern in der Übersetzungsphase,
- Übersichtliche und leichte Lesbarkeit von Programmprotokollen,
- Vereinfachung der Erstellung und Verwaltung der Dokumentation.

2.
Anwenderbedingungen

Für die Assemblierung ist erforderlich:

- Mit dem Editor erfaßter Quelltext in MABS/CABS/SABS 1520 auf Diskette,
- Eine initialisierte und mit dem Bibliotheksservice eröffnete Diskette mit der entsprechenden Bibliothek (RELOCLB, PHASELB) oder mit beiden Bibliotheken.

Verwendete technische Mittel - Maschinenkonfiguration

Das Programm "Assembler" erfordert folgende Mindestausstattung:

- RAM-Arbeitsspeicher mind. 20/20/24 K Byte,
- Bildschirm 1024 oder 1920 Byte,
- Drucker 1152 oder 1157,
- Floppy Disk mit ≥ 1 Laufwerk (2 Laufwerke ist Standard!).

3.
Beschreibung von Aufgabe und Lösung

3.1.
Beschreibung der Aufgabe

Der Assembler hat die Aufgabe, die (mit Hilfe des Editors) erfaßten Quellprogramme in die Modulform oder direkt in die Form einer verschiebbaren Phase zu übersetzen, formale Fehler zu kontrollieren und anzuzeigen sowie die Programme für den Anwender auf Datenträger bereitzustellen. Wahlweise soll der Anwender Drucklisten von Assemblerausdruck, Fehlerprotokoll, oder Bildschirmausschriften erhalten.

3.2. Lösungsbeschreibung

3.2.1. Allgemeines

MBAS/CASS/SASS 1520 (SIEX) arbeiten bei der Übersetzung der in MABS/CABS/SABS 1520 (s. dazu Sprachbeschreibung der Assemblersprache MABS/CABS/SABS 1520), mit dem EDITOR erfaßten Quellprogramme, in 3 Etappen:

1. Kommunikation (Beginndialog: Bediener-MBAS/CASS/SASS)
2. PASS1
3. PASS2

Die Dialogarbeit der Kommunikation erfolgt über Bildschirm: mit alphanumerischer Tastatur werden die zur Assemblerlaufsteuerung benötigten Kommandos eingegeben.

Die Assemblierung erfolgt (intern) in 2 Übersetzungsläufen:

PASS1 - Aufbau der Symboltabelle

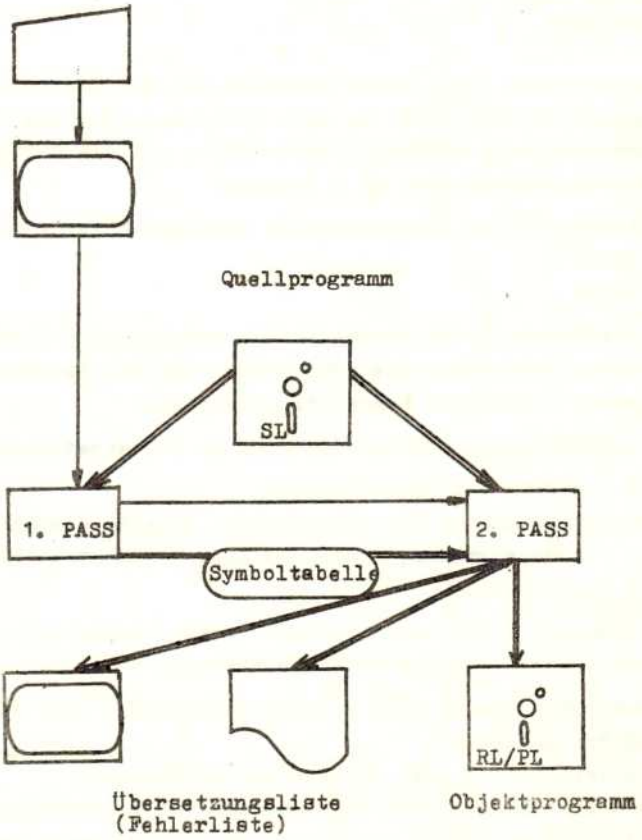
PASS2 - Übersetzung des Quelltextes, Ausgabe des Maschinencodeprogramms, Druck des Assemblerprotokolls/Fehlerprotokolle

Der Anwender erhält nach seiner Vorgabe, außer der entsprechenden Phase oder dem Modul, wahlweise:

- Assemblerausdruck oder Anzeige vollständig oder teilweise auf Bildschirm,
- Fehlerprotokoll oder Anzeige auf Bildschirm,
- Symboltabellen: Druck oder Anzeige auf Bildschirm.

Der Assembler ist ein speicherintensives Programm und arbeitet ständig mit der gesamten geforderten Maschinenkonfiguration (s. erforderliche Anlagenkonfiguration). Eingabe Quelltext - Übersetzung - Ausgabe Maschinencodedruck/Anzeige des Übersetzungstextes ist ein bis zur Endeerkennung ständig fortlaufender Zyklus.

3.2.2. Übersicht über den prinzipiellen Ablauf der Übersetzung und Datenfluß



3.3. Dialogarbeit

3.3.1. Tastatur

Die Bedienung erfolgt über alphanumerische Tastatur. Jede Anweisung wird durch Eingabe eines Steuerzeichens (1-8, A-D) angewählt.
Schreibweise beliebig: Groß- oder Kleinbuchstaben, lateinische Tastatur.
Als Textendetasten werden die Tasten ET1 (oder jede andere Starttaste außer ET2) und ET2 verwendet.

3.3.2. Bildschirm

3.3.2.1. Allgemeines

Die Bildschirmeinheit wird zur Bedienerführung eingesetzt.
Es werden folgende Komponenten angezeigt:

- Eingabezeile,
- Zusammenstellung aller Anweisungen,
- Statuszeile,
- Fehleranzeige.

Es können Bildschirmeinheiten mit 1024 oder 1920 Byte verwendet werden, wobei beim "großen Bildschirm" der Inhalt des 1K-Schirmes in Bildschirmmitte angezeigt wird.

3.3.2.2. Eingabezeile

Die Eingabezeile wird zur Anzeige der vom Bediener über Tastatur eingegebenen Anweisung verwendet. Sie umfaßt 63 Zeichen und ist durch Begrenzungsstriche oberhalb und unterhalb markiert.

3.3.2.3. Zusammenstellung aller Anweisungen

Im unteren Bildschirmteil werden ständig die möglichen Anweisungen angezeigt (Menü). Dabei werden im linken Bildschirmteil die Definitionsanweisungen und im rechten Bildschirmteil die Aktionsanweisungen dargestellt.

Ausnahme:
Eine Ausgabe der Liste oder der Symboltabelle überschreibt den Bildschirm bis auf die Statuszeile. Nach Beendigung dieser Funktion wird der ursprüngliche Bildschirminhalt wiederhergestellt.

3.3.2.4. Statuszeile

Auf der letzten Bildschirmzeile werden alle angewiesenen Definitionen dargestellt.

- Quellbuchname
- Objektbuchname
- Anweisung Liste:
  Assemblerliste/Errorliste,
  Drucker/Bildschirm,
  Zeilenanzahl je Seite,

Listentyp,
Basisadresse für Liste,
Druckgrenzen.

- Zuweisung Objekttext,
- Aktuelle Zeilennummer bei Übersetzung.

3.3.2.5. Fehleranzeige

Fehler, die bei der Abarbeitung einzelner Anweisungen auftreten, werden in einer speziellen Fehlerzeile angezeigt. Dafür rückt die untere "Begrenzung" um eine Zeile nach unten.

Je nach Fehlertyp wird nach Beheben des Fehlers (z.B. Drucker-Fehler) durch Betätigung der ET1-Taste weitergearbeitet oder in den Assemblergrundzustand zurückgegangen.

3.4. Anweisungen

3.4.1. Format

Nach Eingabe der Steuerziffer wird die zugehörige Anweisung in der Eingabezeile angezeigt, sofern sie eine zusätzliche Spezifikation benötigt.

3.4.2. Anweisungsgruppen

Die Funktionen der Assembler werden durch folgende Anweisungsgruppen realisiert:

- Definitions-Anweisungen,
- Aktions-Anweisungen.

Die D e f i n i t i o n s a n w e i s u n g e n dienen
- der logischen Zuordnung der peripheren Geräte zum Assembler,
- der Festlegung des Datenstroms.

Sie sind vor den Aktionsanweisungen auszuführen.

Die A k t i o n s a n w e i s u n g e n dienen
- zum Ausführen des Übersetzungsvorgangs,
- zum Drucken/zur Anzeige des Symbolverzeichnisses,
- zum Rücksprung in den Monitorzustand.

3.5. Beschreibung des Übersetzungsvorgangs

Der Übersetzungsvorgang läuft in 2 Schritten ab (Pässe genannt):

PASS1:

Der PASS1 analysiert die Quellzeile. Je nach Art der Quellzeile wird eine spezielle Behandlung eingeleitet.

Makrobefehl (ohne WAIT), CPU-Befehl ohne Parameter:
Erhöhen Speicherzuweisungszähler laut Mnemoniktabelle.

Makrobefehl (WAIT möglich), CPU-Befehl mit Parameter:
Analyse Parameterteil, Einstellen WAIT-Selektor, Erhöhen Speicherzuweisungszähler laut Mnemoniktabelle und WAIT-Selektor bzw. laut Parameteranalyse.

Pseudobefehl, der Einfluß auf Speicherzuweisungszähler hat:
Analyse Parameterteil,
Erhöhen Speicherzuweisungszähler, laut Analyse Parameterteil.

Kommentarzeilen und Pseudobefehle, die keinen Einfluß auf den Speicherzuweisungszähler haben und keine Eintragung in die Symboltabelle hervorrufen, werden übergangen.

Pseudobefehle, die Eintragungen in Symboltabelle vornehmen:
Analyse Parameterteil und Eintragen der Symbole in Symboltabelle

Danach wird, wenn der Befehl markiert war, die Marke in die Symboltabelle eingetragen.
Der erste Paß wird beendet, wenn die Vorwärtskennung der gerade bearbeitenden Zeile FFFF ist.

PASS2:

Der zweite PASS analysiert ebenfalls die Quellzeilen und prüft deren syntaktische Zulässigkeit. Die Befehle werden in das Objektprogramm transformiert. Für das Programm wird eine Übersetzungsliste erstellt. Der zweite PASS verwendet außer dem Quellprogramm die im ersten PASS aufgebaute "Symboltabelle".

Die Übersetzungsliste kann (außer auf dem Drucker) wahlweise auch über Bildschirm ausgegeben werden. In der Fehlerliste erfolgt die Ausgabe aller fehlerhaften Quellzeilen.

Für die Verarbeitung der Quellzeilen ist folgendes charakteristisch:

- Zerlegung einer Quellzeile in die syntaktischen Einheiten Symbol, Operationscode und Menge von Parametern,
- Überprüfung der Korrektheit der syntaktischen Elemente sowie Ermittlung des ersten erkannten Fehlers für falsche Quellzeilen,
- Aufruf spezieller Routinen zur Verarbeitung der Pseudobefehle, die außer der Syntaxkontrolle folgendes generieren:
  . Steuerinformationen für den Binder (LINK),
  . Druckersteuerinformation,
  . Dateninformation für den Objektcode.

4.
Ein- und Ausgabedaten

4.1.
Eingabedaten

- Operatoreingaben zur Steuerung der Assemblierung durch Tastatur,
- Ein mit dem Editor erfaßtes Quellbuch, in der SL (source library) auf Diskette,
- Diskette mit dem Dienstprogramm "Bibliotheksservice" eröffneter RL (relocatable library) bzw. (phase library).

4.2.
Ausgabedaten

Mit dem Dienstprogramm xxAS 1520 (SIEX) werden dem Anwender folgende Ausgabedaten wahlweise zur Verfügung gestellt:

- Maschinencodeprogramme für das System SIOS in den Formen Modul oder Phase,
- als Druckliste ein Assemblerprotokoll mit Fehlercodeangaben bei formalen Fehlern des Quelltextes (s. Fehlertabelle in den Sprachbeschreibungen),
- eine Fehlerliste, ERROR LIST (EL) mit Fehlercodeangaben und der entsprechenden Quellzeile,
- Druck einer Symboltabelle,
- Bildschirmanzeige
  - als verkürzte Liste ohne Maschinencodeanzeige der übersetzten Quellzeilen
  - als vollständige Liste mit Maschinencodeanzeige und Quelltext,
  - als Fehlerliste,
  - als Symboltabelle.

Anwendungsbeschreibung

Programmbezeichnung: Linkage Editor
Kurzbezeichnung: LINK 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Das Dienstprogramm LINK 1520 (SIEX) ist Bestandteil des Servicekomplexes:
EDITOR - ASSEMBLER - BINDER - BIBLIOTHEKSSERVICE.
Er hat die Aufgabe, verschiebbare Objektprogramme der RL zu einer absoluten oder verschiebbaren, durch das Kommunikationssystem aus der PL ladbaren Phase zu verbinden.

2. Anwendungsbedingungen

Mindestgeräteausstattung

Speicherbedarf: 10 K Byte + Speicher für dynamische Tabellen
≥ 2 K Byte
Externe Geräte: Wahlweise Drucker (log.Adr.OB),
Bildschirm 1024 oder 1920 Byte,
Mindestens ein FD-Laufwerk.

3.
Beschreibung von Aufgabe und Lösung

3.1.
Aufgabe

- Lösen externer Leerbindungen getrennt übersetzter Programme (Programme, Programmteile, Unterprogramme),
- Prüfen der Bibliotheken/Bücher,
- Zusammenfügen verschiebbarer Objektprogramme der RL zu einer absoluten oder verschiebbaren Phase,
- Erzeugung von Überlagerungsstrukturen.

3.2.
Lösung

Der Binder besteht aus 5 Moduln:

- Steuermodul,
- Startmodul,
- Kommunikationsmodul,
- Filehandler,
- Textmodul.

Bei Eingabe aller notwendigen Daten durch den Bediener werden entsprechende Tabellen generiert. Danach wird der erste Paß ausgeführt, der die Tabelle globaler Symbole generiert. Ist ein Autolinklauf gewünscht, so wird anhand aller nicht aufgelösten Symbole ein Suchvorgang in allen aufgefundenen RL ausgelöst. Der 2. Paß löst die lokale Symboltabelle auf und formiert die Phase.
Symbolfehler werden bei einem zugewiesenen Protokollgerät ausgegeben.

4.
Ein- und Ausgangsdaten

Eingabedaten

Bücher der RL, Kommandos des Operators

Ausgabedaten

Buch in der PL,
(Fehlerliste),
(Basispunkttabelle) (wahlweise ob Druck),
(Symboltabelle).

Anwendungsbeschreibung

Programmbezeichnung: Bibliotheksservice
Kurzbezeichnung: LBSV 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Das Dienstprogramm Bibliotheksservice (LBSV) unterstützt den Anwender beim Aufbau und der Pflege von Bibliotheken, die für die Programmspeicherung und -erzeugung notwendig sind.
Vgl. Systemhandbuch, Programmspeicherung, Programmerzeugung, Pkt. 1.5.

2. Anwendungsbedingungen

Es werden initialisierte und geprüfte Disketten verwendet.
Empfohlene Sektorfolge: 01

Technische Mittel:

Konfiguration: ≥ 2 Floppy-Disk-Laufwerke
1 Bildschirm mit 1024 oder 1920 Zeichen
1 Drucker
Speicherbereich: ≥ 20 K Byte

3.
Beschreibung von Aufgabe und Lösung

3.1.
Beschreibung der Aufgabe

Mit dem Dienstprogramm Bibliotheksservice werden folgende Hauptaufgaben realisiert:

- Eröffnen von Bibliotheken,
- Erweitern von Bibliotheken,
- Verkleinern von Bibliotheken,
- Löschen von Bibliotheken,
- Löschen von Quellbüchern, Moduln oder Phasen,
- Drucken von Quellbüchern,
- Drucken von Bibliotheks-Verzeichnissen,
- Umbenennen von Quellbüchern, Moduln oder Phasen,
- Anzeigen von Bibliotheks-Verzeichnissen,
- Kopieren von Quellbüchern, Moduln oder Phasen,
- Kopieren von Bibliotheken,
- Verdichten von Bibliotheken durch Kopieren.

3.2.
Beschreibung der Lösung

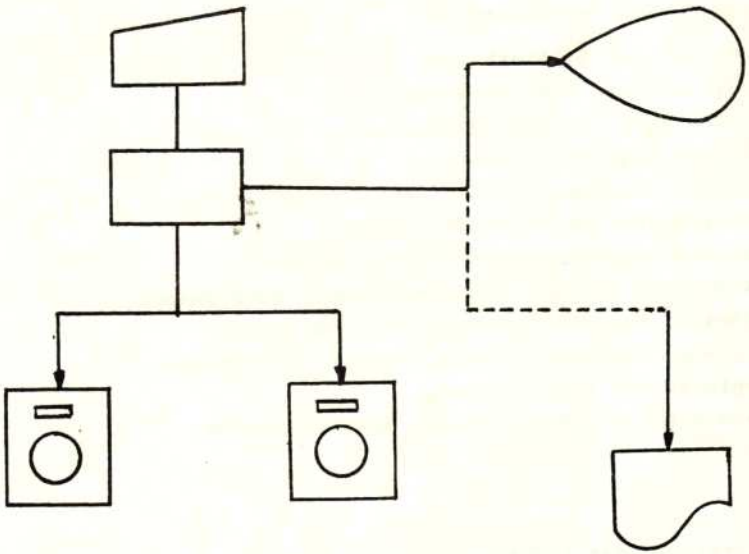
3.2.1.
Dialogarbeit

3.2.1.1.
Allgemeines

Die Arbeit des LBSV erfolgt dialogorientiert. Dabei werden folgende Komponenten einbezogen:

- Tastatur,
- Bildschirm,
- Wahlweise Drucker.

Übersicht:



3.2.1.2.
Tastatur

Die Bedienung erfolgt über die alphanumerische Tastatur. Jede Anweisung wird durch Eingabe eines Steuerzeichens (1-9, A-I) angewählt.
Schreibweise beliebig: Groß- oder Kleinbuchstaben, Lateinische Tastatur.
Als Textendetasten werden die Tasten ET1 und ET2 verwendet.

3.2.1.3. Bildschirm

3.2.1.3.1. Allgemeines

Die Bildschirmeinheit wird zur Bedienerführung eingesetzt. Es werden folgende Komponenten angezeigt:

- Eingabezeile,
- Zusammenstellung aller Anweisungen,
- Statuszeile,
- Fehleranzeige.

Es können die Bildschirmeinheiten mit 1024 oder 1920 Byte verwendet werden, wobei beim "großen" Bildschirm der Inhalt des 1K-Schirmes in Bildschirmmitte angezeigt wird.

3.2.1.3.2. Eingabezeile

Die Eingabezeile wird zur Anzeige der vom Bediener über Tastatur eingegebenen Anweisung verwendet. Sie umfaßt 64 Zeichen und ist durch Begrenzungsstriche im oberen Bildschirmteil markiert. Außerdem werden hier eventuelle Eingabefehler gekennzeichnet. Im Fehlerfall bleibt der Zeileninhalt erhalten.

3.2.1.3.3. Zusammenstellung aller Anweisungen

Im unteren Bildschirmteil werden ständig die möglichen Anweisungen angezeigt. Dabei werden im linken Bildschirmteil die Definitionsanweisungen sowie die Endanweisung dargestellt.

Auf der rechten Bildschirmseite sind alle Serviceanweisungen aufgeführt.

Ausnahme:
Die Funktion "Verzeichnisanzeige" überschreibt das untere Bildschirmfeld mit dem Verzeichnisinhalt. Nach Beendigung dieser Funktion wird der ursprüngliche Bildschirminhalt wiederhergestellt (vgl. Pkt. 3.2.2.4.4.).

3.2.1.3.4. Statuszeile

Auf der letzten Bildschirmzeile werden alle angewiesenen Definitionen dargestellt:
- Druckerzuweisung,
- Definition der Quelldiskette,
- Definition der Zieldiskette,
- Definition des Bibliothekstyps.

3.2.1.3.5. Fehleranzeige

Fehler, die bei der Abarbeitung einzelner Anweisungen auftreten, werden dem Bediener im Errorfeld auf der ersten Bildschirmzeile (durch Unterstreichung Kursormerkmal) augenscheinlich bekanntgegeben.
Es kann jetzt eine neue Anweisung angewählt werden (Eingabe eines Steuerzeichens).
Enthält die fehlerhafte Anweisung einen Variablenteil, so besteht die Möglichkeit, die Korrektur der Anweisung vorzunehmen.
Darstellung der Fehleranzeige s. Anlage 3.

3.2.1.4. Drucker

Während der Arbeit mit LBSV besteht die Möglichkeit, den Drucker oder Zusatzdrucker zu beliebigen Zeitpunkten zuzuordnen (vgl. Pkt. 3.2.2.3.1.). Dann erfolgt die Protokollierung aller aufgeführten Anweisungen.
Die Druckerzuweisung ist darüber hinaus auch Voraussetzung für die Ausführung der Funktion "PRINT" (vgl. 3.2.2.4.2.).

3.2.2. Anweisungen

3.2.2.1. Format

Nach Eingabe der Steuerziffer wird die zugehörige Anweisung in der Eingabezeile angezeigt.
Bei Anweisungen, die einen variablen Teil enthalten (Steuerzeichen 1, 2, 3, 4, 8, A, C, F, H), können dann wahlweise ein oder mehrere Operanden (z.B. Buchnamen) eingegeben werden. Mehrere Operanden sind durch Komma voneinander zu trennen.

Beispiel: DELET PROG 1, PROG 2, PROG 3

Es wird kein Endekennzeichen für die Anweisung eingegeben. Die Abarbeitung erfolgt so, daß die Anweisung nach vollständiger Eingabe und Betätigen der Starttaste (ET1 oder ET2) ausgeführt wird. Während der Abarbeitung bleibt die Eingabezeile erhalten. Nach Ausführung der Anweisung erlischt die Eingabe. Eine neue Anweisung ist möglich.

3.2.2.2.

Anweisungsgruppen

Die Funktionen des LBSV werden durch folgende Anweisungsgruppen realisiert:

- Definitions-Anweisungen,
- Service-Anweisungen,
- End-Anweisung.

Die Definitionsanweisungen dienen:

- der logischen Zuordnung der peripheren Geräte für den LBSV,
- der Festlegung des Type der jeweils behandelnden Bibliothek,
- der Eröffnung, Erweiterung und Verkleinerung von Bibliotheken.

Sie sind den Serviceanweisungen zeitlich vorgelagert.
Serviceanweisungen sind vom Benutzer frei wählbar. Sie umfassen folgende Anweisungen:

- Löschen von Büchern, Moduln oder Phasen,
- Löschen von Bibliotheken,
- Drucken von Büchern,
- Drucken von Verzeichnissen,
- Umbenennen von Büchern, Moduln oder Phasen,
- Anzeigen von Verzeichnissen,
- Kopieren von Büchern, Moduln oder Phasen,
- Kopieren von Bibliotheken.

Die End-Anweisung dient der Beendigung des Programms.

3.2.2.3. Definitions-Anweisungen

3.2.2.3.1. ASSGN-Anweisung

Mit der ASSGN-Anweisung wird die logische Gerätezuordnung für das LBSV eingestellt, d.h. es können Drucker und Floppy-Disk-Laufwerke zugewiesen werden.

Die Anweisung

ASSGN␣P 1, LL

spezifiziert den 1. Drucker,

ASSGN␣P 2, LL

spezifiziert den Zusatzdrucker.
Die Zuweisung bleibt bis zur nächsten Anweisung ASSGN P 1 bzw. ASSGN P 2 erhalten.
Mit LL werden die Anzahl der auf den verwendeten Leporello-papier druckbaren Zeilen je Seite angegeben.
Es gilt: $6 < L < 73$.

Wird LL nicht eingegeben, so wird die Standardzeilenzahl 72 eingestellt.
Vor der Druckerzuweisung muß das Leporello in Grundstellung gebracht werden.
Wurde eine dieser Anweisungen ausgeführt, so werden ab diesem Zeitpunkt alle nachfolgenden Anweisungen protokolliert. Sie ist unbedingt erforderlich, um die Funktion PRINT auszuführen. Tritt während der Ausführung des Drucks ein Hardware-Fehler auf, so wird die Zuweisung des Druckers bzw. Zusatzdruckers aufgehoben.
Die Zuweisung wird in der Statuszeile angezeigt.
Mit LL = Ø kann die Zuweisung aufgehoben werden.

Hinweis:

Diese Anweisungen sind der mit der Monitorfunktion ASN eingestellten Geräteverteilung untergeordnet.

Mit den Anweisungen

ASSGN␣1, VOL - NAME [,Z]
ASSGN␣2, VOL - NAME [,Z]

werden die Floppy-Disk-Laufwerke und Datenträger zugeordnet. Dabei bedeuten:

1 = Quelldiskette
2 = Zieldiskette

Über die Quelldiskette werden die Funktionen NEWLB und alle Servicefunktionen realisiert.
Für die Funktion COPY ist zusätzlich die Definition der Zieldiskette erforderlich.
Mit VOL-Name wird der Datenträgername von Quell- oder Zieldiskette spezifiziert. Damit sind Verwechslungen von Disketten ausgeschlossen.
Durch die Angabe Z (4-7) kann eine Zuordnung von Quell- und Zieldiskette zu den Floppy-Disk-Laufwerken, d.h. deren logischen Geräteadressen, erfolgen.
Fehlt diese Angabe, so erfolgt die Standard-Zuweisung:

Quelldiskette -- LW ≙ log. Geräte-Adresse 4
Zieldiskette -- LW ≙ log. Geräte-Adresse 5

Auch die Laufwerksauswahl ist der über die Monitorfunktion ASN festgelegten Geräteverteilung untergeordnet.
Jede Anweisung bleibt solange erhalten, bis eine neue ASSGN-Anweisung erfolgt, die sich auf die Quell- bzw. Zieldiskette bezieht.
Vor der Abarbeitung jeder anderen Anweisung (außer LB Typ und END) wird die durch ASSGN eingestellte Datenträgerdefinition geprüft.
Die Zuweisungen werden in der Statuszeile angezeigt.

3.2.2.3.2. LBTYP-Anweisung

Durch die Anweisung LBTYP wird in der weiteren Arbeit mit dem LBSV der zu behandelnde Bibliothekstyp definiert.

Dabei bedeuten:

LBTYP␣S	-	Quelltextbibliothek	SL
LBTYP␣R	-	Modulbibliothek	RL
LBTYP␣P	-	Phasenbibliothek	PL

Der Bibliothekstyp gilt für Quell- und Zieldiskette. Er bleibt bis zur nächsten LBTYP-Anweisung erhalten. In der Statuszeile wird der definierte Bibliothekstyp angezeigt.

3.2.2.3.3. NEWLB-Anweisung

Für die Definition, Erweiterung oder Verkleinerung von Bibliotheken wird die Anweisung

NEWLB␣TT

benutzt.
TT gibt die Anzahl der für die gesamte Datei (einschließlich Verzeichnis) vorgesehenen Spuren auf der Quelldiskette an ($1 \leq TT \leq 74$).
Wird TT nicht eingegeben, so gilt der Standardfall 74 Spuren.
Der Bibliothekstyp ist vorher durch die Anweisung LBTYP festzulegen und die Quelldiskette durch ASSGN␣1,... zu definieren.

Eröffnen Bibliothek

Befindet sich noch keine Bibliothek auf der Diskette, wird geprüft, ob die vorgesehene Anzahl von Spuren zusammenhängend frei ist (Berücksichtigung aller auf der Spur 0 definierten

Dateien). Ist dies erfüllt, wird die Bibliothek initialisiert (vgl. Systemhandbuch Programmspeicherung / Programmerzeugung).

Erweitern Bibliothek

War auf der Diskette diese Datei schon vorhanden, wird geprüft, ob TT größer als die von der Bibliothek bisher belegten Spuren ist. Bevor die Differenzanzahl von Spuren in Anschluß an die Datei geschrieben werden kann, wird geprüft, ob die für die Erweiterung vorgesehenen Spuren unmittelbar nach der Bibliothek durch keine andere Datei belegt sind. Der ursprüngliche Dateninhalt der Bibliothek bleibt erhalten.

Verkleinern Bibliothek

Bei der Verkleinerung einer Bibliothek wird geprüft, ob TT größer als die von der Datei bisher belegte Spurenanzahl ist. Der Dateninhalt der Bibliothek bleibt erhalten.

3.2.2.4. Service-Anweisungen

3.2.2.4.1. Delet-Anweisung

Die Anweisung

DELET NAME 1, [NAME 2 ...]

wird zum Löschen von Büchern, Moduln oder Phasen benutzt. Es können ein oder mehrere Bücher, Moduln oder Phasen gelöscht werden. Mehrere Namen werden durch Komma getrennt. Bibliothekstyp und Quelldiskette sind vor "DELET" durch die Definitions-Anweisungen festzulegen.

Vor der Ausführung der Anweisung wird geprüft, ob alle aufgeführten Namen im Bibliotheksverzeichnis enthalten sind. Ist ein Name nicht enthalten, wird die gesamte Anweisung nicht ausgeführt. Mit der folgenden Fehlermeldung werden alle im Verzeichnis nicht enthaltenen Namen markiert.

Sind alle Namen in der Bibliothek enthalten, werden sie im Verzeichnis gelöscht. Dadurch entstehen Lücken in Verzeichnis und Datenteil der Bibliothek (Verdichtung s. COPY-Anweisung).

Mit der Anweisung

DELET ALL

wird die gesamte Bibliothek gelöscht (HDR 1 in DDR 1 gewandelt).

3.2.2.4.2. PRINT-Anweisung

Folgende PRINT-Anweisungen sind zugelassen:

PRINT NAME 1 [, NAME 2 ...]
PRINT DIC
PRINT ALL

Durch die Anweisung "PRINT NAME 1 [, NAME 2 ...] kann der Inhalt von Büchern einer Quellbibliothek gedruckt werden. Moduln oder Phasen werden nicht gedruckt.
Für die Prüfung der Buchnamen im Verzeichnis gelten die gleichen Hinweise wie bei der DELET-Anweisung.
Mit der Anweisung PRINT DIC können Verzeichnisse aller Bibliotheksarten gedruckt werden.

Hinweis:
Bei gelöschten Büchern, Moduln oder Phasen wird kein Name gedruckt.

Die Anweisung PRINT ALL erzeugt die vollständige Druckliste einer Quelltextbibliothek, d.h. Verzeichnis und Inhalt aller Bücher.

Drucker, Bibliothekstyp und Quelldiskette müssen vorher durch Definitions-Anweisungen festgelegt worden sein.

3.2.2.4.3. RENAM-Anweisung

Die RENAM-Anweisung

RENAM NAME 1, NAME 2,

wird zum Umbenennen eines Buchs, Moduls oder einer Phase verwendet.
Dabei bedeuten:

NAME 1 - alter Name
Name 2 - neuer Name

Es muß gesichert sein, daß NAME 1 im Verzeichnis enthalten und NAME 2 nicht enthalten ist.

Bibliothekstyp und Quelldiskette müssen definiert worden sein.

3.2.2.4.4. DSPLY-Anweisung

Mit der Anweisung

DSPLY DIC

wird das Verzeichnis der jeweils definierten Bibliothek auf dem unteren Bildschirmteil angezeigt.
Ist der Füllstand des Verzeichnisses größer als der für die Anzeige verwendete Bildschirmbereich, kann durch Betätigen der Taste ET1 die Anzeige des nächsten Verzeichnisteils vorgenom-

men werden. Die Taste ET2 beendet die Anzeige des Verzeichnisses. Bibliothekstyp und Quelldiskette müssen definiert worden sein.

Hinweis:
Gelöschte Bücher, Moduln oder Phasen werden als Leerzeichen im Namensfeld dargestellt.

3.2.2.4.5. COPY-Anweisung

Das Kopieren von Büchern, Moduln oder Phasen wird durch die Anweisung

COPY␣NAME 1, NAME 2 ...

realisiert.
Vor Ausführung der Anweisung wird geprüft, ob Quell- und Zieldiskette auf verschiedenen Laufwerken mit unterschiedlichen Datenträgernamen definiert wurden und auf diesen Disketten jeweils eine Bibliothek des zugewiesenen Typs vorhanden ist.
Darüber hinaus erfolgt die Prüfung, ob alle zu kopierenden Bücher, Moduln oder Phasen in der Quellbibliothek enthalten und in der Zielbibliothek noch nicht enthalten sind. Vor dem Kopieren wird der Füllstand von Verzeichnis und Datenteil in bezug auf die aufzunehmenden Bücher, Moduln oder Phasen geprüft.
Lücken (gelöschte Namen) im Verzeichnis der Quelldiskette werden übergangen, bestehende Lücken im Verzeichnis der Zieldiskette bleiben erhalten.
Quellbücher werden verdichtet übertragen.
Das Datenteil jeder Zielbibliothek wird lückenlos fortlaufend gefüllt. Das Verzeichnisteil der Zieldiskette wird fortgeschrieben.

Hinweis:
Beim Kopieren von Quellbüchern wird ständig geprüft, ob das Ende der Bibliothek erreicht wird.

Die Anweisung

COPY␣ALL

bewirkt das Kopieren der gesamten Quelltextbibliothek auf die Zielbibliothek in verdichteter Form.

Für den Vergleich der beiden Verzeichnisse in bezug auf gleiche Namen gilt die obige Darstellung.

Handelt es sich bei der Zielbibliothek um eine fortgeschriebene Bibliothek, so wird diese um den Inhalt der Quellbibliothek (verdichtet) fortgeschrieben.
Ist die Zielbibliothek neu (noch kein Buch, Moduln oder Phase enthalten), so stellt das Duplikat im Ergebnis der Kopierfunktion die verdichtete Quellbibliothek dar.

3.2.2.5. END-Anweisung

Zur Beendigung des Dienstprogramms wird die Anweisung

END

benutzt.
Die Programmwiederholung ist mit der Monitorfunktion

NEW ET1, ET1

möglich.

3.2.3.
Fehlerbehandlung

Fehlerbehandlung und Fehlermeldungen erfolgen nach der Richtlinie zur Gestaltung der Bildschirmarbeit und der Fehlerbehandlung in Dienst- und Hilfsprogrammen.

Hinweis zur Organisation der Programmerzeugung:
Für Programmkorrekturen und eventuelle Zerstörungen der gespeicherten Informationen auf der Diskette wird empfohlen, Duplikate auf verschiedenen Disketten zu führen.

4.
Ein- und Ausgabedaten

4.1.
Eingabedaten

Unter Eingabedaten werden die
- Bibliotheken, gespeichert auf Diskette
  und
- Anweisungen, eingegeben über Tastatur

verstanden.

4.2.
Ausgabedaten

Unter Ausgabedaten versteht man die erzeugten, fortgeschriebenen oder gelöschten Bibliotheken.

Anwendungsbeschreibung

Programmbezeichnung: Makroprogramm - Testsystem
Kurzbezeichnung: MTES 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

MTES dient zur Unterstützung des Tests von Programmen, die in der Makrobefehlssprache MABS 1520 geschrieben sind. Mikroprogrammteile (in U880-CPU-Sprache) können eingelagert sein.

2. Anwendungsbedingungen

Alle Anzeigen und Eingaben von MTES erfolgen auf den ersten oder vier letzten Bildschirm-Zeilen (werden nicht gerettet). Das zu testende Makroprogramm darf nicht den Speicherbereich von MTES belegen.

Technische Mittel

Speicherbereich

- MTES - gesamt: 6 K Bytes Anwenderbereich
- MTES (im Abarbeitungsbereich): 5 K Bytes

MTES-gesamt belegt stets den Speicherbereich ab Anfang Anwenderbereich. Das eigentliche (verschobene, absolut adressierte) MTES kann jeden beliebigen Bereich des Anwenderbereiches (außer Bildschirm) belegen.

Konfiguration

- FD für Programmeingabe MTES bzw. MBK,
- Bildschirm 1024 oder 1920 Byte,
- Drucker (wenn Protokollierung-Druck gewünscht),
- Weitere Geräte entsprechend zu testendem Makroprogramm,
- FD bzw. KMB bei Ausgabe auf Phasenbibliothek.

3. Beschreibung der Aufgabe und der Lösung

Mit MTES können in der Makrobefehlssprache MABS 1520 geschriebene Programme (Makroprogramme) dynamisch getestet werden.

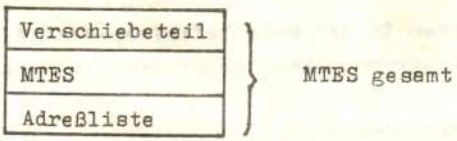
Folgende Funktionen werden realisiert:

- Kontrollierte Abarbeitung des zu testenden Makroprogramms bis zu einer Stop-Stelle,
- Anzeige bzw. Druck von Befehlszähler, Befehlsmnemonik, Befehlscodierung,
- Anzeige bzw. Druck des Inhalts eines frei wählbaren Speicherbereiches (1 - 16 Byte),
- Anzeige oder Druck eines zusammenhängenden Speicherbereiches beliebiger Länge,
- byteweise Anzeige von Speicherbereichen oder byteweise Eingabe in Speicherbereiche,
- Ausgabe eines Speicherbereiches als feste Phase in eine Phasenbibliothek auf Floppy Disk oder Magnetbandkassette.

MTES ist ein CPU-Programm, das Makroprogrammteile (physische Ein-/Ausgaben) enthält. Nach dem Laden kann es mit einem Verschiebeteil an jede beliebige Stelle des Anwender-RAM-Bereiches transportiert werden.
MTES befindet sich gleichzeitig mit dem zu testenden Programm (Anwenderprogramm) im Hauptspeicher.
Zur Bedienerkommunikation werden wahlweise die ersten oder letzten vier Bildschirmzeilen genutzt (werden nicht gerettet).

3.1. Verschiebung

Das verschiebbare Gesamtprogramm besteht aus drei zusammenhängenden Teilen:



und wird an den Anfang des Anwenderbereiches geladen (AAW).
Durch Ansprung der Adresse AAW wird der Verschiebeteil aktiviert.
Dort wird die Adresse des Anfangs der letzten Bildschirmzeile ermittelt und der Programmname angezeigt.
Es wird die Eingabe der absoluten Speicheradresse erwartet, ab der MTES gespeichert werden soll.
Nach der Eingabe werden mit Hilfe der Adreßliste in MTES die relativen Adressen in absolute umgewandelt. Dabei enthält die Adreßliste die Adressen der L-Teile der in MTES zu modifizierenden Adressen.

Nach Beendigung der Adressenänderung wird ein Teilprogramm, das den Transport von MTES in den eingegebenen Bereich reali-

siert, in die letzte Bildschirmzeile transportiert und vom Programm selbst aktiviert. Nach dem Transport von MTES befindet sich das System im Monitor-Status.
Es kann das zu testende Programm geladen und MTES durch Sprung an seine eingegebene Anfangsadresse aktiviert werden.

Wird MTES immer an einen festen Speicherbereich geladen, genügt eine einmalige Verschiebung in diesen Bereich. Nach Ausgabe auf Programmspeicher kann dann MTES stets direkt geladen werden (Wegfall von Verschiebeteil und Adreßliste).

3.2. Programmbeginn

Nach Anwahl von MTES erfolgt die Anzeige des Programmnamens mit Versionsnummer, dem von MTES nach der Verschiebung belegten Speicherbereich sowie dem Kommunikationsgerät Bildschirm und reserviertem Bildschirmbereich (F ... erste vier Zeilen (first), L ... letzte vier Zeilen (last)).
Der Bildschirmbereich kann wahlweise übernommen oder geändert werden (L ←→ F).

3.3. Grundschleife

Nach Festlegung des reservierten Bildschirmbereiches befindet sich MTES in der Grundschleife. Eine Rückkehr zum Programmbeginn ist möglich.

In der Grundschleife erfolgt die Anzeige von

T ->,

und es wird die Eingabe eines Kommandos erwartet.

Alle Eingaben werden mit der Starttaste ET1 beendet, bei Betätigen von ET2 wird die Eingabe abgebrochen und in die Grundschleife zur Eingabewiederholung zurückgekehrt. Ein Kommando besteht aus dem Kommando-Namen und Kommando-Parameter. Zur Trennung der Kommando-Parameter wird von MTES ein Minus-Zeichen angezeigt.

Folgende Kommandos sind in MTES möglich:

Kommandoname	Bedeutung
D M P	Anzeige/Druck zusammenhängender Speicherbereiche
M E M	byteweise Speicherarbeit
T E P	Testpunktfestlegung
S T P	Stop-Punkt-Steuerung
B I B	Ausgabe in Phasenbibliothek

3.4. Anzeige/Druck zusammenhängender Speicherbereiche (DMP)

Die Kdo-Parameter werden in folgender Reihenfolge eingegeben:

- Protokolliergerät:
  - D ... Bildschirm
  - P ... Drucker
- Datendarstellung:
  - X ... Hexadarstellung
  - C ... zeichenweise im ISO-7-Bit-Code Bereich 20H ... 7FH, sonst Punkt)
  - A ... X und C (nur bei Druck)
- Anfangsadresse

3.4.1.
Anzeige auf dem Bildschirm

In Zeile 1 des Bildschirmanzeigebereichs wird eine Überschriftzeile (niedrigstes Halbbyte der Speicheradresse) angezeigt, in den Zeilen 2 - 4 der Inhalt von 48 zusammenhängenden Speicherplätzen.
Bei Betätigen von ET1 wird der angezeigte Bereich um jeweils 16 Byte nach oben verschoben.
Nach Betätigen von ET2 wird zur Grundschleife zurückgekehrt.

3.4.2.
Druck

Zu Beginn des Speicherausdruckes erfolgt der Druck einer Überschriftszeile, diese Überschriftszeile wird nach Erhöhung des höheren Adreßbytes um 1 wiederholt. Dabei erfolgt ein Formularvorschub um 2 Zeilen. Jeweils am Zeilenbeginn werden die höheren 3 Adreß-Halbbytes in Hexa-Darstellung gedruckt.
Bei Datendarstellung A erfolgt der Druck hexa- und zeichenweise jeweils in einer Zeile.

3.5.
Byteweise Speicherarbeit

Die Kdo-Parameter werden in folgender Reihenfolge eingegeben:

- Datendarstellung für Werteingabe: X ... hexadezimal
  C ... zeichenweise im ISO-7-Bit-Code
- Adresse (direkt oder symbolisch)
- Wert bzw. Leertaste (ET1)

3.5.1.
Direkte Adresse

Nach Eingabe der Speicheradresse wird sofort ihr Inhalt angezeigt (hexadezimal). Durch Eingabe eines Wertes wird dieser alte Inhalt überschrieben und die Adresse um 1 erhöht. Dabei wird von MTES geprüft, ob eine Eingabe in die geschützten Bereiche

a) ØØØØH - (AAW - 1) (AAW ... Anfangsadresse Anwenderbereich)

b) Speicherbereich von MTES

erfolgen soll und dann Fehler angezeigt.

Wird nur ET1 betätigt (Leertaste), so bleibt der angezeigte Inhalt erhalten, es erfolgt nur eine Hochzählung der Adresse. Nach ET2 kann eine neue Adresse eingegeben oder durch nochmaliges Betätigen von ET2 zur Grundschleife zurückgekehrt werden.

3.5.2.
Symbolische Adresse

Für bestimmte Speicherbereiche von MTES, die Informationen über das Anwenderprogramm enthalten, kann über symbolische Adressen zugegriffen werden. Dies kann zur Vorgabe von Anfangszuständen bzw. zum Ändern oder zur Anzeige während des Testlaufs genutzt werden.

Symbolische Adresse	Bedeutung
P C L	L-Teil Makrobefehlszähler
P C H	H-Teil Makrobefehlszähler
N	Starttastenregister N
A	Starttastenregister A
B C	Bedingungscode
S T	Statusbyte
K L	L-Teil für Selektorregister K
K H	H-Teil für Selektorregister K
K R L	L-Teil Selektorregister KR
K R H	H-Teil Selektorregister KR

Nach der Eingabe der symbolischen Adresse wird sofort ihr aktueller Inhalt hexadezimal angezeigt.
Es ist die Eingabe eines neuen Inhalts möglich.
Anschließend kann, wie auch bei Betätigen der Leertaste (ET1), eine neue (symbolische oder direkte) Adresse eingegeben werden.

3.6.
Testpunktfestlegung (TEP)

Durch die Testpunktfestlegung erfolgt eine Spezifizierung der Protokollierung während der schrittweisen Abarbeitung des zu testenden Programms.
Die Kdo-Parameter werden wie folgt eingegeben:

- Protokolliergerät: P ... Drucker
  D ... Bildschirm
- Testpunktspezifikation: Protokollierung erfolgt bei Angabe von:

  EØ bei jedem Befehl

AD <Befehlszählerstand>	bei jedem Befehl, der bei dem angegebenen Befehlszählerstand beginnt
OC <Operationscode>	bei jedem Befehl mit dem angegebenen Operationscode
MC <Anfangsadresse Bereich> <Bereichslänge>	bei Änderung des Inhalts des angegebenen Speicherbereichs
JP	bei jedem Sprungbefehl

- Protokollierbereich: Anfangsadresse (Länge 1 - 16)

oder Leertaste, wenn kein Protokollierbereich gewünscht wird.

Jeder abgearbeitete Makrobefehl wird wie folgt protokolliert: Befehlszähler vor Abarbeitung - Befehlsmnemonik - Befehlscodierung - Protokollierbereich (falls vorhanden) nach Abarbeitung.
Auf dem Bildschirm wird jeder abgearbeitete Befehl (unabhängig von der Testpunktspezifikation) angezeigt.
Wurde der Befehl durch 'EX' modifiziert bzw. durch 'MLF' oder 'MLS' verändert, wird dies durch die Anzeige von 'E' bzw. 'M' nach der Befehlscodierung gekennzeichnet.

3.7.
Stop-Punkt-Steuerung (STP)

Mit diesem Kommando wird die Abarbeitung des zu testenden Programms gesteuert. Die Kdo-Parameter werden in folgender Reihenfolge eingegeben:

- Betriebsart:	Stop im zu testenden Anwenderprogramm bei:
EO	nach jedem Befehl (Schrittbetrieb)

AD <Befehlszählerstand>	nach Abarbeitung des Befehls, der beim angegebenen Befehlszählerstand beginnt
OC <Operationscode >	nach jedem Befehl mit dem angegebenen Operationscode
MC <Anfangsadresse Bereich> <Bereichslänge>	nach jedem Befehl, der den Inhalt des angegebenen Bereiches ändert
JP	nach Ausführung eines Sprungbefehls

- Neuer Befehlszählerstand für abzuarbeitendes Programm oder Leertaste, wenn angezeigter aktueller Befehlszählerstand nicht geändert werden soll.

Nach der Eingabe des SIP-Kommandos wird die Abarbeitung des zu testenden Programms gestartet und der durch den Befehlszähler adressierte Makrobefehl abgearbeitet.

Beachte:

- Eingelagerte Mikrobefehlsfolgen (durch CALC oder EOI aktiviert) werden im Echtzeitbetrieb abgearbeitet.
  Stop-Punkte, die sich auf Mikrobefehlsfolgen beziehen, werden ignoriert. Durch den Befehl SOP übernimmt MTES automatisch wieder die Steuerung.
  Testunterstützung für Mikrobefehlsfolgen wird durch das Testhilfsprogramm TMIK gewährt.
- Der Befehl TIMER und der Vorbefehl W werden von MTES wegen der zeitverzögerten Abarbeitung des Anwenderprogramms nicht abgearbeitet. Dies wird durch die Anzeige 'N' nach der Befehlscodierung gekennzeichnet.
- Die Befehle OFF und EOP führen zu einem Stop-Punkt, unabhängig von der Stop-Punkt-Spezifikation.

Vor Abarbeitung eines Befehls werden die Speicherplätze für den Bedingungscode, Starttastencode, Status-, K- und KR-Register aus einem Rettungsbereich geladen und ihr aktueller Inhalt nach Abarbeitung dorthin transportiert.
Während der Abarbeitung bleibt das STP-Kommando auf dem Bildschirm angezeigt. Der aktuelle Befehlszählerstand nach Befehlsabarbeitung wird jeweils in dieser Zeile angezeigt, so daß bei Sprungbefehlen die Ausführung oder Nichtausführung sofort ersichtlich ist.

Bei anliegenden Eingabebefehlen wird deren Befehlsmnemonik (ohne Befehlscodierung, Protokollierbereich) angezeigt.

Nach Befehlsausführung und Protokollierung erfolgt bei Erreichen eines STP-Punktes ein Stop zur visuellen Kontrolle.
Nach Betätigen von ET1 wird die nächste Befehlsfolge abgearbeitet. Durch ET2 wird zur Grundschleife zurückgekehrt, es können Programmkorrekturen vorgenommen, ein neuer Protokollierbereich gewählt, der aktuelle Speicherinhalt protokolliert und neue Anfangswerte definiert werden u.a.
Durch STP wird die Abarbeitung des zu testenden Programms an der unterbrochenen oder an einer beliebigen anderen Stelle fortgesetzt.

3.8.
Ausgabe in Phasenbibliothek (BIB)

Durch Ausführung dieses Kommandos wird ein Speicherbereich (z.B. korrigiertes Anwenderprogramm) als feste Phase in eine auf FD oder MBK befindliche Phasenbibliothek (PL) ausgegeben.

Die Kdo-Parameter werden in folgender Reihenfolge eingegeben:

- Buchname (Programmname) für auszugebende Phase
- Anfangsadresse) des als Phase auszugebenden
- Endadresse) Speicherbereiche

- Ausgabegerät: FD ... Floppy Disk
  CT ... Magnetbandkassette
- Logische Geräteadresse des Ausgabegerätes

4.
Eingangs- und Ausgangsdaten

Eingangsdaten

- Kommandos durch Tastatur

Ausgangsdaten

- Anzeige Speicherinhalt
- Druck/Anzeige Speicherbereiche
- Druck/Anzeige abgearbeiteter Makrobefehl
  (Abarbeitungsprotokoll)
- Feste Phase auf FD oder MBK

Anwendungsbeschreibung

Programmbezeichnung: Testhilfe für Mikrobefehlsfolgen (U880)
Kurzbezeichnung: TMIK 1520 (SIEX) VØ.1

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1.
Zweckbestimmung des Programms

TMIK bietet eine effektive Unterstützung bei einem dynamischen Test von CPU-Programmen auf einem Rechner, der den Mikroprozessor U880 als CPU enthält und mit dem Makrointerpreter MINT 1520 arbeitet.

Folgende Funktionen werden von TMIK realisiert:

- Abarbeitung von CPU-Programmen im Echtzeit-Betrieb bis zu einer beliebigen Stop-Adresse. Eingelagerte Makro-Programmteile werden dabei normal abgearbeitet.
- Schrittweise Abarbeitung von CPU-Programmen mit der Anzeige von
  - Befehlsadresse,
  - Befehlsmnemonik,
  - Befehlscodierung,
  - CPU-Register-Inhalt nach Befehlsabarbeitung.
- Beliebige Manipulation des Inhalts der CPU-Register PC, SP, IX, IY, A, F, B, C, D, E, H, L,
- Anzeige des Inhalts des Flag-Bytes als Bitmuster,

- Vertauschung Registersatz mit Hilfsregistersatz,
- Byteweise Anzeige von Speicherbereichen bzw. Eingabe in Speicherbereiche,
- Hexadezimaler Druck von zusammenhängenden Speicherbereichen.

2. Anwendungsbedingungen

Alle Anzeigen und Eingaben von TMIK erfolgen auf den letzten drei Bildschirmzeilen (werden nicht gerettet).
TMIK und das zu testende Programm dürfen sich beim Test nicht überlagern.

Speicherbedarf:

- TMIK-gesamt: ØD76H Byte (3446)
- TMIK (im Arbeitsbereich): ØB28 Byte (2856)

TMIK-gesamt belegt stets den Bereich ab Anfang Anwenderbereich. Das eigentliche TMIK kann jeden beliebigen Bereich des Anwenderbereichs (außer Bildschirm-Bereich) belegen.

Geräte

- BAB1 (BAB2)
- Drucker, wenn DP-Komponente benutzt werden soll,
- Geräte zur Eingabe von TMIK und dem zu testenden Anwenderprogramm,
- Weitere Geräte entsprechend dem zu testenden Anwenderprogramm.

3.
Beschreibung von Aufgabe und Lösung

3.1.
Allgemeines

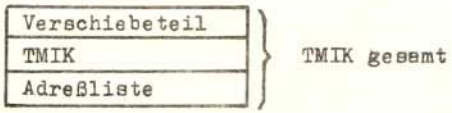
TMIK ist ein CPU-Programm; das Makroprogrammteile (physische Ein-/Ausgaben) enthält. Nach dem Laden kann es durch ein Verschiebeteil an jede beliebige Stelle des Anwender-RAM-Bereiches transportiert werden.
TMIK befindet sich gleichzeitig mit dem zu testenden Programm (Anwenderprogramm) im Hauptspeicher.
TMIK benutzt zur Kommunikation mit dem Tester die letzten drei Zeilen des Bildschirmbereiches (werden nicht gerettet). Dabei dienen die ersten beiden Zeilen zur Anzeige, wobei als Überschriftenzeile in der ersten Zeile die Bezeichnung der CPU-Register angezeigt wird:

PC	SP	IX	IY	A	F	B	C	D	E	H	L

In der zweiten Zeile steht ihr jeweiliger aktueller Inhalt hexadezimal.
In der dritten Anzeigezeile erfolgt die Kommandoeingabe bzw. Befehlsanzeige bei Schrittbetrieb, Anzeige des Stop-Punktes für Echtzeitbetrieb und Anzeige des Speicherinhalts bei byteweiser Eingabe/Anzeige.
Nach Aktivierung von TMIK befindet sich das System in der Grundschleife, in der die aktuellen CPU-Registerinhalte angezeigt werden und eine Kommando-Eingabe erwartet wird. Nach Ausführung des jeweiligen Kommandos wird in die Grundschleife zurückgekehrt.

3.2.
Verschiebung

Das verschiebbare Gesamtprogramm besteht aus drei zusammenhängenden Teilen:



und wird an den Anfang des Anwenderbereiches geladen (AAW). Durch Ansprung der Adresse AAW wird der Verschiebeteil aktiviert.
Dort wird die Adresse des Anfangs der letzten Bildschirmzeile ermittelt und der Programmname angezeigt. Es wird die Eingabe der absoluten Speicheradresse erwartet, ab der TMIK gespeichert werden soll.
Nach der Eingabe werden in TMIK die relativen Adressen mit Hilfe der Adreßliste in absolute umgewandelt. Dabei enthält die Liste die Adressen der L-Teile der in TMIK zu modifizierenden Adressen.
Nach Beendigung der Adressenänderung wird ein Teilprogramm, das den Transport von TMIK in den angegebenen Bereich realisiert, in die letzte Bildschirmzeile transportiert und vom Programm selbst aktiviert. Nach dem Transport von TMIK befindet sich das System im Monitor-Status.
Es kann das zu testende Programm geladen und TMIK durch Sprung an seine eingegebene Anfangsadresse aktiviert werden. Wird TMIK immer an einen festen Speicherbereich geladen, genügt eine einmalige Verschiebung in diesen Bereich. Nach Ausgabe auf Programmspeicher kann dann TMIK stets direkt geladen werden (Wegfall von Verschiebeteil und Adreßliste).

3.3.
Echtzeit-Betrieb (GO)

Es wird hier ein Programmteil, beginnend beim eingestellten PC-Stand bis zu einer eingestellten Stop-Adresse, im Echtzeitbetrieb abgearbeitet. Der CPU-Befehl, der bei der angegebenen Stop-Adresse beginnt, wird dabei nicht mehr abgearbeitet.
Die eingestellten CPU-Register-Inhalte werden als Anfangswerte zu Beginn der Abarbeitung verwendet. Bei Erreichen des Stop-Punktes werden die aktuellen Inhalte nach Abarbeitung des letzten Befehls angezeigt. Der Echtzeit-Betrieb wird durch das Einspeichern eines RST 7 (ØFFH) in die durch den Stop-Punkt adressierte Speicherstelle realisiert. Nach Erreichen des Stop-Punktes wird der ursprünglich vorhandene Speicherinhalt wieder hergestellt. Das bedeutet, daß bei vorzeitigem Abbruch der Echtzeitabarbeitung das FF-Byte nicht eliminiert wird.

3.4.
Schritt-Betrieb (TR)

Beginnend mit der aktuellen PC-Einstellung und den CPU-Registerinhalten als Anfangswert wird ein CPU-Programm schrittweise abgearbeitet. Nach der Abarbeitung eines Befehls werden

Befehlsadresse - Befehlsmnemonik - Codierung

und die neuen CPU-Register-Inhalte angezeigt. Es kann wahlweise mit dem nächsten Befehl im Schritt-Betrieb fortgesetzt oder zur Grundschleife zurückgekehrt werden.
TMIK enthält für den Schrittbetrieb einen Analyseteil, der die Mnemonik des anliegenden CPU-Befehls und dessen Länge ermittelt. Die Abarbeitung erfolgt unter TMIK-Regie. Dabei wird die Ausführung aller Sprünge simuliert.

Nicht zugelassene CPU-Befehle (IN, OUT, I-/R-Register-Befehle) und ungültige Operationscodes werden gekennzeichnet und nicht ausgeführt. RST-Befehle werden richtig angezeigt, aber ebenfalls nicht ausgeführt.

3.5. Register-Manipulation

Der Inhalt der CPU-Register PC, SP, IX, IY, A, F, B, C, D, E, H, L, kann beliebig eingestellt werden. Dazu wird die angegebene Mnemonik des Registers eingegeben und der neue Wert hexadezimal. Nach der Eingabe wird der neue Inhalt des Registers sofort angezeigt.

3.6. Flag-Anzeige (FF)

Anstelle der hexadezimalen Anzeige des Flag-Byte-Inhalts wird er als Bitmuster unter der Überschrift

SZ. H. PNC

angezeigt.

3.7. Registeraustausch (EX)

Der Inhalt der CPU-Register A F B C D E H L wird mit dem Inhalt der entsprechenden Hilfsregister getauscht. Durch ein nochmaliges EX-Kommando kann die Vertauschung rückgängig gemacht werden.

3.8.
Byteweise Speicherarbeit (M)

Nach Angabe einer beliebigen Speicheradresse wird deren Inhalt hexadezimal angezeigt. Es kann ein neuer Wert eingegeben werden oder ohne Wertveränderung zur nächsthöheren Adresse übergegangen werden.

3.9.
Speicherausdruck (DP)

Der Inhalt eines zusammenhängenden Speicherbereiches wird hexadezimal ausgedruckt, wobei am Zeilenanfang die Adresse des ersten Bytes gedruckt wird. Ein Teil der letzten Bildschirmzeile wird als Druckpuffer benutzt, eine Druckzeile besteht aus 53 Druckpositionen.

4.
Ein- und Ausgangsdaten

Eingangsdaten

- Kommandos durch Tastatur,
- Codierungen des zu testenden Anwenderprogramms.

Ausgangsdaten

- Anzeige Speicherinhalte, CPU-Register-Inhalte,
- Druck Speicherbereiche,
- Anzeige abgearbeiteter CPU-Befehle.

Anwendungsbeschreibung

Programmbezeichnung: Konvertieren einer sequentiellen (aufsteigend sortierten) Datei in eine gestreute (direkt adressierte) Datei

Kurzbezeichnung: KONV

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Erstellen einer gestreuten (direkt adressierten) Datei aus einer aufsteigend sortierten sequentiellen Datei.
Die sequentielle Datei muß sich auf einer Diskette befinden.
Die gestreute Datei wird ebenfalls auf eine Diskette ausgegeben.

2. Anwendungsbedingungen

- Die Quelldatei muß sich auf einer Diskette befinden.
- Sie muß aufsteigend sortiert sein, darf keine gelöschten Datensätze enthalten. Schlüsselbegriffe dürfen nicht mehrfach auftreten.
- Der Kennsatz der Zieldatei muß auf der Zieldiskette stehen.
- Die maximale logische Satzlänge beträgt 1Ø24 Byte.
- Die Datei muß ungeblockt und ungespant sein.

- Der Schlüsselbegriff muß rein numerisch sein und kann maximal 5Ø Stellen aufweisen (zusammenhängend).
- Die logische Satzlänge muß für beide Dateien gleich sein.

Technische Mittel

- Speicherbedarf: 5 K Byte
- Externe Geräte: 1 Bildschirm 1Ø24 oder 192Ø Byte
  1 bzw. 2 Floppy-Disk-Laufwerke

3. Beschreibung von Aufgabe und Lösung

3.1. Beschreibung der Aufgabe

Das Programm KONV realisiert die satzweise Übernahme von der sequentiellen Datei und die Ausgabe in die gestreute Datei. Es wird auf Doppelbelegungen der rein numerischen Schlüsselbegriffe geprüft. Bei auftretenden Doppelbelegungen wird das Programm abgebrochen. Weiterhin wird auf Einhaltung der Sortierfolge geprüft. Treten Sortierfehler auf, so erfolgt ebenfalls Abbruch des Programms. Der erste Schlüsselbegriff definiert den ersten Satz in der gestreuten Datei. Alle weiteren Schlüsselbegriffe beziehen sich in ihrer Satzfolge auf den ersten Schlüsselbegriff. Treten Lücken im Schlüsselsystem auf, so werden diese Lücken als gelöscht markierte Datensätze in die gestreute Datei ausgegeben.
An Hand der Differenz: Erster Schlüsselbegriff zu gesuchtem Schlüsselbegriff + 1 kann mit READD direkt auf den entsprechenden Datensatz zugegriffen werden. Es entfällt das sequentielle Suchen des Datensatzes. Die Anwendung dieser Dateiform ist nur geeignet bei relativ "geschlossener" Schlüsselsystematik (Wertevorrat mit wenig Lücken) und einer max. Anzahl von

1924 Sätzen mit einer Satzlänge von 128 Byte je logischer Satz.

3.2. Beschreibung der Lösung

Nach dem Laden des Programms und der Menü-Anzeige wird der Bediener aufgefordert, das logische Gerät und den Dateinamen der Quelldatei einzugeben. Bei Eingabefehlern bzw. "Datei nicht vorhanden" erfolgt eine entsprechende Fehlermeldung. Danach wird der Bediener aufgefordert, das logische Gerät und den Dateinamen der Zieldatei einzugeben. Bei Eingabefehlern bzw. "Datei nicht vorhanden" erfolgt eine entsprechende Fehlermeldung.

Im Anschluß muß der Bediener die Position des ersten Zeichens des Schlüsselbegriffes in Satz eingeben. Es erfolgt eine Bereichsprüfung. Bei Eingabefehler erfolgt eine Fehlermeldung. Danach muß der Bediener die Länge des Schlüsselbegriffs eingeben. Ebenfalls erfolgt eine Prüfung auf zulässigen Wertevorrat. Bei Eingabefehler erfolgt eine Fehlermeldung. Danach stoppt das Programm an einen Haltepunkt zur Überprüfung aller Eingaben. Nach Quittierung für die richtigen Eingaben erfolgt die Konvertierung.
Alle anliegenden Schlüsselbegriffe werden auf dem Bildschirm angezeigt. Treten Fehler auf, so erfolgt eine entsprechende Fehlermeldung.
Das Ende der Konvertierung wird durch "END" angezeigt.

4.
Ein- und Ausgangsdaten

4.1.
Eingangsdaten

- Diskette mit einer aufsteigend sortierten sequentiellen Datei,
- Bedienereingaben.

4.2.
Ausgangsdaten

- Diskette mit erzeugter gestreuter Datei,
- Fehlermeldung.

Anwendungsbeschreibung

Programmbezeichnung: Berechnung der Prüfziffer
Kurzbezeichnung: SCNG 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1.
Zweckbestimmung des Programms

Berechnung der Prüfziffer SELF-CHECKING NUMBER GENERATOR.

2.
Anwendungsbedingungen

Das Programm ist ein eigenständiges Programm und benötigt Bildschirm und Drucker.
Der Druck erfolgt ab Position Ø5 im Format A4 hoch (66 Druckpositionen).
Das Programm wird auf die Anfangsadresse Anwenderbereich geladen und auch ab dort gestartet.
Als Randbedingung für die Parameter gilt:
Der Modul M darf kein ganzzahliges Vielfaches der Gewichtung der Prüfziffer sein.
Der Prüfrest ist < Modul.

3.
Beschreibung der Aufgabe und der Lösung

Aufgabe

Mit Hilfe des Programms wird die Prüfziffer mit einem eingetasteten Ordnungsbegriff (Zahl) errechnet.

Dazu werden vorher durch Eingabe als Parameter spezifiziert:

1. Modul (Modulo),
2. Prüfrest (Remainder of checking),
3. Stelle der Prüfziffer (Position of checking digit),
4. Quersummebildung JA/NEIN (Calculate traverse sum Y/N),
5. Stellengewichte W 16 ... W Ø1 (weight factors of positions).

Die Berechnung der Prüfziffern für fortlaufende Ordnungsbegriffe ist möglich. Die errechnete Prüfziffer wird, sofern sie einstellig ist, allein und in die Zahl eingefügt gedruckt. Die verwendbaren Zahlen werden fortlaufend gezählt. Errechnet sich eine zweistellige Prüfziffer, so wird dieser Ordnungsbegriff als nicht verwendbar gekennzeichnet. Die Prüfziffernberechnung/Zahlenprüfung beruht auf folgender Prüfmethode:

$$\sum_{i=1}^{16} \frac{St_i \cdot W_i}{M} = X + DR$$

mit St_i = Wert der Stelle i
W_i = Gewichtung der Stelle i
M = Modul
X = Ganzzahliger Quotient
DR = Divisionsrest

Der sich errechnende Divisionsrest muß mit dem angegebenen Prüfrest übereinstimmen.

Das Programm ermöglicht auch die Prüfziffernberechnung nach der Methode, bei der anstelle der Teilprodukte $St_i \cdot W_i$ nur die Quersummen dieser Teilprodukte in die Rechnung eingehen (Anwendung vorwiegend bei Modul 10).

Lösung

Auf der Grundlage der einzugebenden Parameter Modul, Prüfrest, Stelle der Prüfziffer, Quersummenbildung und Stellengewichte sowie des eingetasteten Ordnungsbegriffes wird für die betreffende Zahl die PZ errechnet. Dabei wird zunächst anstelle der noch unbekannten PZ die Ziffer Ø eingesetzt.

Der 1. Schritt ist die Ermittlung des Divisionsrestes (DR) nach der Formel

$$X + DR = \frac{\sum_{i=1}^{16} St_i \cdot W_i}{M}$$

Der ganzzahlige Quotient X kann für die weitere Rechnung unberücksichtigt bleiben. Ist DR = PR (Prüfrest), ergibt sich die PZ zu Ø. Ansonsten gilt es, die PZ nach folgender Formel zu berechnen:

$$PZ = \frac{n \cdot M + PR - DR}{GPZ}, \quad n = 1, 2, 3, \ldots, m$$

GPZ = Gewichtung der PZ

Dabei ist n solange zu verändern, bis das Ergebnis (PZ) ganzzahlig ist.

Bei M > 10 errechnen sich auch zweistellige Prüfziffern, die sich aber nicht an einer Dezimalstelle einfügen lassen. Die betreffende Zahl kann nicht verwendet werden und wird im Druckprotokoll mit dem Text

Checking digit needs two positions,
NUMBER cannot be used!

gekennzeichnet.

Bei Eingabe der Zahl mit Taste ET1 wird automatisches Weiterzählen erreicht.

4. Ein- und Ausgangsdaten

Eingangsdaten

Parametereingaben für
- Modul 1 ... 15,
- Prüfrest 0 ... 14,
- Stelle der Prüfziffer 1 ... 16,
- Quersummenbildung Y/N,
- Stellengewichte 0 ... 14.

Eingaben der Zahlen, für die die Prüfziffer zu berechnen ist.

Ausgangsdaten

Druckliste, die neben den Eingangsdaten die errechneten Prüfziffern und die Zahlen mit eingefügten Prüfziffern enthält.

Anwendungsbeschreibung

Programmbezeichnung: Anzeige / Inhaltsdruck Sektorinhalt Diskette
Kurzbezeichnung: DINH 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1. Zweckbestimmung

DINH dient als Testhilfe für den Programmierer bei der Arbeit mit dem physischen Format 128 Byte je Sektor.

2. Anwendungsbedingungen

- Bildschirm 1024 oder 1920 Byte,
- Arbeitsbereich RAM mind. 2 K Byte,
- Drucker (wahlweise),
- 1 Laufwerk Floppy disk.

3. Beschreibung von Aufgabe und Lösung

Mit dem Programm DINH kann der Inhalt von Sektoren hexadezimal und alphanumerisch angezeigt bzw. gedruckt werden.

Es wird die aktuelle Spur- und Sektornummer angezeigt bzw. gedruckt.
Die Anzeige erfolgt mit der Eingabe von Spur (TT)/Sektor (SS) und ET1, Anzeige des nächsten Sektors nur ET1.
Der Druck eines Sektors erfolgt analog der Anzeige, jedoch ist die Taste ET2 zu betätigen.
Wird ein Sektor mit DEL-Maske gelesen, erfolgt Anzeige bzw. Druck "MARKED SECTOR".
Die Eingabe von "END" führt zum Abbruch des Programms mit Sprung in den Monitor.
Der Inhalt eines Sektors wird auf dem Bildschirm durch 8 Zeilen angezeigt bzw. auf 8 Zeilen gedruckt.
Programmstart ohne Diskette im Laufwerk führt zum Lesefehler "ERROR READ DISK, DRIVE 04".
Kann ein Sektor nicht gelesen werden, so wird Lesefehler "ERROR READ" angezeigt bzw. gedruckt.
Ein Druckerfehler führt zur Anzeige "ERROR PRINTER".
Zeichen der Codespalten Ø und 1 des Hexacodes werden alphanumerisch als Punkt angezeigt oder gedruckt.
Die Codespalten Ø8 - 15 werden in äquivalente ISO-Zeichen für alphanumerische Anzeige/zum Druck gewandelt.

4.
Ein- und Ausgangsdaten

Eingangsdaten

Spur(TT) und Sektor(SS).

Ausgangsdaten

Sektorinhalt angezeigt oder gedruckt.

Anwendungsbeschreibung

Programmbezeichnung: Anzeige Inhalt Datei/Kassette
Kurzbezeichnung: DIKA 1520 (SIEX) V 1.Ø

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung

Mit dem Programm DIKA kann der Inhalt einer Datei satz- bzw. blockweise auf Bildschirm angezeigt werden.

2. Anwendungsbedingungen

Voraussetzung sind Kassetten mit SIMPLE-, BASIC- oder COMPACT-Dateien.

Anlagenkonfiguration:

- ≥ 12 K Speicher,
- 1 Kassettenlaufwerk,
- 1 Bildschirm mit 1024 oder 1920 Byte.

3.
Beschreibung von Aufgabe und Lösung

Das Programm ermöglicht folgende Funktionen:

- Anzeige des Dateikennsatzes bei COMPACT-Dateien,
- Anzeige eines ausgewählten Satzes der Datei hexadezimal und/oder alphanumerisch,
- Anzeige des letzten Satzes der Datei.

Nach Programmaufruf und -start erscheint auf dem Bildschirm das Start-Menü.

D r Anwender gibt ein:

1.	Log. Geräteadresse	D,E
2.	Anzeigeform	A - alphanumerisch H - alphanumerisch und hexadezimal
3.	Dateiname	S - SIMPLE B - BASIC ≤ 8 Zeichen - COMPACT Nach Eingabe des Namens wird der Dateikennsatz angezeigt.
4.	Dateinummer	Ø1,...,99 - SIMPLE und BASIC Eingabe übergangen - COMPACT
5.	Satznummer	ØØØØ1,...,65535 99999 = letzter Satz der Datei

Auf der 15. bzw. 23. Zeile erscheinen nach erfolgten Eingaben im Menü und mit Anzeige des ersten gewünschten Satzes:

SIMPLE, BASIC oder Dateiname	Dateinummer (bei S oder B)	Aktuelle Satznummer

Nach Anzeige der ersten Seite eines Satzes stoppt das Programm an einer Eingabe.

Nach Leerstart + wird die nächste Seite bzw. der nächste Satz, nach Leerstart - wird der vorhergehende Satz angezeigt.
Nach Eingabe einer max. 4stelligen Nummer wird die Satznummer um diese Zahl erhöht und der entsprechende Satz angezeigt.
Soll ein bestimmter Satz gelesen werden, muß die entsprechende Satznummer eingegeben werden (bei Satznr. < 5 Stellen mit Vornullen eintasten).
Mit ET2 Rückkehr zum Start-Menü.

Fehlermeldungen

Die entsprechenden Ausschriften erscheinen auf der sogenannten Fehlerzeile gemäß der Richtlinie zur Fehlerbehandlung bei Fehlern während Abarbeitung des Menüs oder auf der 16. bzw. 24. Bildschirmzeile bei Fehlern während der Anzeige.

4.
Ein- und Ausgangsdaten

Datensätze der entsprechenden Datei

Bildschirmgröße	64 x 16	80 x 24
A		
Zeichen/Zeile	64	80
max. Zeichen/Seite	896	1760
H		
Zeichen/Zeile	16	16
(links hexadezimal,		
rechts alphanumer.)	224	352

Anwendungsbeschreibung

Programmbezeichnung: Druck / Anzeige RAM/ROM-Speicherinhalt
Kurzbezeichnung: DMPM 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Das Programm dient dazu, einen Speicherbereich
- in den Grenzen von .. bis ..,
- alphanumerisch und hexadezimal,
- auf dem Bildschirm anzuzeigen.
  und wahlweise
- zu drucken.

Der Speicherinhalt wird nicht verändert. Das Programm ist zur Testunterstützung vorgesehen.
Siehe auch Teil "DUMP" des der Testunterstützung dienenden Programms MTES!

2. Anwendungsbedingungen

- Fehlerhafte Eingaben führen zu erneutem Eingabebeginn.
- Laden als verschiebbare Phase auf eine vom Anwender gewünschte Adresse außerhalb des zu druckenden Bereiches.

- Bei Alphadruck werden die Zeichen H'ØA', 'ØB', 'ØC', 'ØD' und 'Ø8', die an sich Steuerfunktionen am Drucker realisieren, als Space (H'2Ø') verarbeitet.

3.
Beschreibung von Aufgabe und Lösung

3.1.
Beschreibung der Aufgabe

Das Programm soll
nach Auswahl einer Beginnadresse einen Speicherinhalt alphanumerisch und hexadezimal anzeigen und wahlweise drucken. Dabei sind nicht durckbare Zeichen als Space zu interpretieren.

3.2.
Lösungsbeschreibung

Die Aufgabe wird durch ein Programm in Makroschreibweise gelöst, das Mikro-UP nutzt.

4.
Ein- und Ausgangsdaten

4.1.
Eingangsdaten

- Adresse von / bis
- Auswahl Anzeige / Druck

4.2.
Ausgangsdaten

- Druckliste mit dem gedruckten Inhalt des gewählten Speichers.

Anwendungsbeschreibung

Programmbezeichnung: Buchtransformation MEOS ----► SIOS
Kurzbezeichnung: BKTR 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1. Zweckbestimmung

Das Dienstprogramm BOOK-TRANSFORMATION (BKTR) dient zum Transformieren von Büchern des Typs S, O(V) und P des Betriebssystems MEOS in die Quell-, Objekt- und Phasenbibliotheken des Betriebssystems SIOS.

2. Anwendungsbedingungen

Es werden initialisierte und geprüfte Disketten verwendet, auf denen die entsprechenden Bibliotheken eingerichtet sind.

Technische Mittel

Konfiguration:	2 Floppy-Disk-Laufwerke
	1 Bildschirm
	(1024 oder 1920 Zeichen)
Speicherbereich:	5 K Byte Anwenderbereich

3.
Beschreibung der Lösung

3.1.
Beschreibung der Aufgabe

3.1.1.
Dialogarbeit

3.1.1.1.
Allgemeines

Die Arbeit von BOOKT erfolgt dialogorientiert. Dabei werden folgende Komponenten einbezogen:
- Tastatur,
- Bildschirm.

3.1.1.2.
Tastatur

Die Bedienung erfolgt über die alphanumerische Tastatur. Durch die Eingabe werden die entsprechenden Aufforderungen von BKTR erfüllt. Jede Aufforderung wird mit einer einzigen Eingabe realisiert. Schreibweise beliebig: Groß- und Kleinbuchstaben, lateinische Tastatur.
Als Textendetasten werden ET2 und ET1 (oder jede beliebige andere Starttaste, Wirkung wie ET1) verwendet.

3.1.1.3.
Bildschirm

Allgemeines

Der Bildschirm wird zur Bedienerführung eingesetzt. Es werden folgende Komponenten angezeigt:
- Eingabezeile,
- Zusammenstellung aller Aufforderungen und die Syntax der Antworten,
- Fehleranzeige.

Es können die Bildschirmeinheiten 1024 Zeichen oder 1920 Zeichen verwendet werden, wobei beim "großen" Bildschirm der Inhalt des 1 K-Schirmes in Bildschirmmitte angezeigt wird.

Eingabezeile

Die Eingabezeile wird zur Anzeige der von BKTR formulierten Aufforderung und zur Eingabe der Antwort des Bedieners verwendet. Sie ist durch Begrenzungsstriche im oberen Bildteil markiert.
Darüber hinaus werden eventuelle Eingabefehler gekennzeichnet.

Zusammenstellung aller Aufforderungen und die Syntax der Antworten

Im unteren Bildschirmteil werden ständig die möglichen Aufforderungen von BKTR angezeigt. Zusätzlich wird die Syntax der Antwort dargestellt. Durch Unterstreichung wird die erfolgte Antwort in diesem Teil gekennzeichnet.

3.1.2. Fehleranzeige

Fehler, die bei der Abarbeitung der Antworten auf die einzelnen Aufforderungen auftreten, werden in der folgenden Zeile nach der Eingabezeile (Verschieben der Begrenzungslinie um eine Zeile nach unten) durch *** ERROR * <Fehlernummer >angezeigt.

3.1.3. Aufforderungen

Die Folge der Aufforderungen wird durch BOOKT vorgegeben. Der Bediener kann sie nur in sofern vorgeben, als er durch Beendigung einer Antwort mit ET2 an deren Anfang zurückkehrt.

Besonderheit:
Wird die erste Aufforderung mit ET2 beendet, so wird das Programm abgebrochen und in den Monitorgrundzustand zurückgekehrt. Liegt die erste Aufforderung an, können sämtliche Disketten gewechselt werden, da die Laufwerke nur in diesem Zustand entriegelt sind.

Aufforderung 1:

LBTYP:

Hier wird die Eingabe des Typs der SIOS-Bibliothek gefordert, in die Bücher zu transformieren sind. Es wird S, R oder P erwartet, womit entsprechend die Quell- (SOURCELB), Objekt- (RELOCLB) oder Phasenbibliothek (PHASELB) spezifiert sind.

Aufforderung 2:

MEOS DISK AT:

Hier wird die Eingabe der logischen Gerätenummer (4/5/6/7) des Laufwerkes, in das die Diskette mit der MEOS-Bibliothek eingelegt wurde, erwartet. Durch Komma getrennt ist der VOL-Name (beim Einrichten der Diskette mit der NAME-Funktion als NAME spezifiziert) einzugeben, der zur Kontrolle für das Einlegen der richtigen Diskette dient.

Aufforderung 3:

SIOS DISK AT:

Hier wird die Eingabe der logischen Gerätenummer (4/5/6/7) des Laufwerkes, in das die Diskette mit der in LBTYP spezifizierten SIOS-Bibliothek eingelegt wurde, erwartet. Durch Komma getrennt ist der VOL-Name einzugeben, der zur Kontrolle für das Einlegen der richtigen Diskette dient.

Aufforderung 4:

BOOKNAMES:

Hier können die Namen der Bücher (durch Komma getrennt) eingegeben werden, die man zu überspielen wünscht. Dabei wird eine Kontrolle auf syntaktische Richtigkeit der Buchnamen (entsprechend MEOS) vorgenommen. Ebenso wird kontrolliert, ob ein Buch mit dem angegebenen Namen in der MEOS-Bibliothek vorhanden und in der SIOS-Bibliothek nicht vorhanden ist.

Beim Überspielvorgang wird das jeweils gerade zu bearbeitende Buch unterstrichen.

Buchnamen Typ S:

- bestehen aus 2 oder 4 Zeichen
- 2 Zeichen: 1. Zeichen Buchstabe
  2. Zeichen Buchstabe oder Ziffer

Es wird das Buch mit der höchsten Versionsnummer überspielt, Als Buchname in der SL gelten die beiden angegebenen Zeichen.

- 4 Zeichen: wie <2 Zeichen> + 2 Ziffern Versionsnummer. Es wird das Buch mit der entsprechenden Versionsnummer überspielt. Als Buchname in der SL gilt der vollständige Buchname (mit Versionsnummer).

Buchname Typ R:

Es gilt das gleiche wie bei Typ S. Werden Bücher in die RL überspielt, müssen in der MEOS-Bibliothek Bücher vom Typ V und O unter dem gleichen Namen vorliegen.

Besonderheiten beim Übertragen von MEOS-Büchern des Typs V und O in die RL von SIOS 1520:

Die nachgenannten Besonderheiten ergeben sich aus der unterschiedlichen Art der Abspeicherung von Objekttexten, der unterschiedlichen Art des Objekttextes und der unterschiedlichen Sprachkonzeption der BS MEOS und SIOS.

a) Sollen mehrere überspielte Module mit LINK (SIEX 1520) zu einer Phase gebunden werden, so müssen sämtliche Symbole unterschiedlich sein. Es gibt also keine globalen und lokalen Symbole, sondern nur noch globale.
D.h. z.B.:
Im Programm A1 besteht kein Unterschied zwischen M und A2.A oder A3.M. Während des Bindelaufes wird eine Doppeldefinition erkannt.

b) Insgesamt dürfen je Modul nicht mehr als 254 Symbole gebraucht werden, da SIEX nur globale Symbole in das Symbolverzeichnis aufnimmt, und ESID-Nr. zwischen 1 und 255 verteilt, MEOS hingegen nimmt alle im Programm erwähnten Symbole (lokale und externe, durch PN. gekennzeichnet) in das SV auf und verteilt ESID-Nr. zwischen 1 und 65535.

Da der Relativassembler die Symbole in der Reihenfolge ihrer Erwähnung ins SV im Programm aufnimmt, wäre die Grenze der Gesamtsumme von 254 dadurch zu überschreiten, daß alle externen Symbole unter den ersten 254 das erste Mal erwähnt werden.

c) Ausdrucke in MEOS Quellprogrammen, die ein externes Symbol in Verbindung mit relativen lokalen oder externen Symbolen haben, müssen in zu übernehmenden Programmen verboten werden.

d) H ( ), L ( ) von externen Symbolen bzw. Ausdrücken mit externen Symbolen müssen verboten werden.

Werden diese Besonderheiten bei der Programmierung von CPU-Programmen auf dem MRES beachtet, ist es möglich, durch Überspielen auf die RL (SIOS) und Binden mit LINK (SIEX 1520) verschiebbare Programme für SIEX 1520 zu erhalten.

Buchnamen Typ P:

Bestehen aus 1 bis 6 Zeichen. Dabei ist das 1. Zeichen ein Buchstabe, die folgenden jedoch müssen Buchstaben oder Ziffern sein. Als Buchname in der PL gelten nur die ersten 5 Zeichen.

3.1.4. Fehlerbehandlung

Fehlerbehandlung und Fehlermeldungen werden durch BKTR selbst übernommen. Das betrifft insbesondere:

- Syntaktische Analyse der Antworten,
- Logische Prüfung,
- Übertragung von Floppy Disk nach Floppy Disk.

Fehlernummern mit Interpretation und Behandlung sind in der Fehlerzeile angezeigt.

Dieser Fehler ist zu quittieren
- Wird mit ET2 quittiert, so erfolgt ein Neuanlauf ab Aufforderung 1.
- Wird mit ET1 quittiert, so erfolgt:
  a) bei Aufforderung 2,3 --- Neuaufforderung
  b) bei Aufforderung 4 --- Fortsetzung der Abarbeitung.

Hinweis zur Organisation des Transformationsvorganges:
Um spezielle Bücher vor Zerstörung auf der Diskette zu schützen, wird empfohlen, sich nach dem Transformationsvorgang Duplikate auf anderen Disketten mit Hilfe von LBSER (Library-Service-Programm) herzustellen.

3.2. Beschreibung der Lösung

Das Programm BKTR wurde als CPU-Programmpaket im Betriebssystem MEOS geschrieben.
Nur zur Handhabung der externen Geräte wurden die Makrobefehle:
ENTC, POS, WAIT, ONF, EXIT, WRITE, READ, READD
benutzt.

Auf Simultanarbeit wurde verzichtet.

Das Programm BOOKT besteht aus 4 Programmen:

VA: Globale Variable, einschließlich erforderlicher DDB und DCB
KO: Kommunikationsteil
FH: Filehandler
BK: Rahmenprogramm

4.
Ein- und Ausgabedaten

Eingabedaten

Unter Eingabedaten wird die MEOS-Bibliothek
- Auf Diskette gespeichert,
- Antworten, eingegeben über Tastatur,

verstanden.

Ausgabedaten

Unter Ausgabedaten werden fortgeschriebene SIOS Bibliotheken verstanden (um transformierte Bücher erweitert).

Anwendungsbeschreibung

Programmbezeichnung: Phase von Diskette nach Kassette
Kurzbezeichnung: PHAK 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1. Zweckbestimmung

Das Dienstprogramm "Phase von Diskette nach Kassette" (PHAK) unterstützt den Anwender beim Kopieren von Büchern aus der Phasenbibliothek auf das Kassettenmagnetband.

2. Anwendungsbedingungen

Es werden initialisierte und geprüfte Disketten verwendet. Empfohlene Sektorfolge Ø3.

Technische Mittel

Konfiguration: ≥ 1 Kassettenlaufwerk
≥ 1 Floppy-Disk-Laufwerk
1 Bildschirm mit 1024 oder 1920 Zeichen
1 Drucker

Speicherbereich: ≥ 20 K Byte

3.
Beschreibung von Aufgabe und Lösung

3.1.
Beschreibung der Aufgabe

Mit dem Dienstprogramm "Phasen von Diskette nach Kassette" werden folgende Hauptaufgaben realisiert:

- Initialisieren und Vorbereiten von Kassettenmagnetbändern (KMB),
- Drucken von Bibliotheksverzeichnissen,
- Anzeigen von Bibliotheksverzeichnissen,
- Kopieren von einzelnen Phasen von Disketten nach KMB,
- Kopieren von Phasenbibliotheken,
- Löschen des Bandes durch Initialisieren.

3.2.
Lösungsbeschreibung

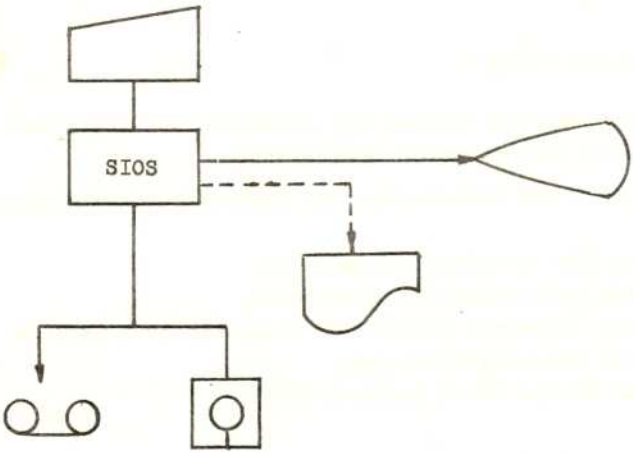
3.2.1.
Dialogarbeit

3.2.1.1.
Allgemeines

Die Arbeit des PHAK erfolgt dialogorientiert. Dabei werden folgende Komponenten einbezogen:

- Tastatur,
- Bildschirm,
- Wahlweise Drucker.

Übersicht:



3.2.1.2.
Tastatur

Die Bedienung erfolgt über die alphanumerische Tastatur. Jede Anweisung wird durch Eingabe eines Steuerzeichens (1 - 7, A - F) angewählt.
Schreibweise beliebig; Groß- oder Kleinbuchstaben, Lateinische Tastatur. Als Textende werden die Tasten ET1 und ET2 verwendet.

3.2.1.3. Bildschirm

3.2.1.3.1. Allgemeines

Die Bildschirmeinheit wird zur Bedienerführung eingesetzt. Es werden folgende Komponenten angezeigt:

- Eingabezeile,
- Zusammenstellung aller Anweisungen,
- Statuszeile,
- Fehleranzeige.

Es können die Bildschirmeinheiten mit 1024 oder 1920 Byte verwendet werden, wobei beim "großen Bildschirm" der Inhalt des 1 K-Schirmes in Bildschirmmitte angezeigt wird.

3.2.1.3.2. Eingabezeile

Die Eingabezeile wird zur Anzeige der Anweisung verwendet, die der Bediener über Tastatur eingegeben hat. Sie umfaßt 64 Zeichen und ist durch Begrenzungsstriche im oberen Bildschirmteil markiert.
Darüber hinaus werden hier eventuelle Eingabefehler gekennzeichnet. Im Fehlerfalle bleibt der Zeileninhalt erhalten. Die Berichtigung erfolgt auf der Folgezeile (vgl. Pkt. 3.1.3.5.).

3.2.1.3.3. Zusammenstellung aller Anweisungen

Im unteren Bildschirmteil werden die möglichen Anweisungen ständig angezeigt. Dabei werden im linken Bildschirmteil die Definitionsanweisungen sowie die Endanweisung dargestellt. Auf der rechten Bildschirmseite sind alle Serviceanweisungen aufgeführt.

Ausnahme:
Die Funktion "Verzeichnisanzeige" überschreibt das untere Bildschirmfeld mit dem Verzeichnisinhalt. Nach Beendigung dieser Funktion wird der ursprüngliche Bildschirminhalt wieder hergestellt (vgl. Pkt. 3.2.4.2.).

3.2.1.3.4. Statuszeile

Auf der letzten Bildschirmzeile werden alle angewiesenen Definitionen dargestellt:
- Druckerzuweisung,
- Definition der Quelldiskette,
- Definition des Zieles KMB,
- Definition des Bibliothekstypes.
  (vgl. Pkt. 3.2.3.1. und 3.2.3.2.)

3.2.1.3.5. Fehleranzeige

Fehler, die bei der Abarbeitung einzelner Anweisungen auftreten, werden dem Bediener im Errorfeld auf der ersten Bildschirmzeile (durch Unterstreichung Kursormerkmal) augenscheinlich bekanntgegeben.

Es kann jetzt eine neue Anweisung angewählt werden (Eingabe eines Steuerzeichens).
Enthält die fehlerhafte Anweisung einen variablen Teil, so besteht die Möglichkeit, nach Drücken der Starttaste ET1 oder ET2 (ohne Steuerzeichen) die Korrektur der Anweisung vorzunehmen.
Mit erfolgter Korrektureingabe der richtigen Anweisung auf der Folgezeile erlischt die Fehleranzeige.

3.2.1.4. Drucker

Während der Arbeit mit PHAK besteht die Möglichkeit, den Drucker oder Zusatzdrucker zu beliebigen Zeitpunkten zuzuordnen (vgl. Pkt. 3.2.3.1.). Dann erfolgt die Protokollierung aller ausgeführten Anweisungen. Die Druckerzuweisung ist darüber hinaus auch Voraussetzung für die Ausführung der Funktionen "PRINT" (vgl. Pkt. 3.2.4.2.).

3.2.2. Anweisungen

3.2.2.1. Format

Nach Eingabe der Steuerziffer wird die zugehörige Anweisung in der Eingabezeile angezeigt.
Bei Anweisungen, die einen variablen Teil enthalten (Steuerzeichen 1,2,3,4,E), können dann wahlweise ein oder mehrere Operanden (z.B. Buchnamen) eingegeben werden. Mehrere Operanden sind durch Komma voneinander zu trennen.

Beispiel: COPY PROG 1, PROG 2, PROG 3.

Es wird kein Endekennzeichen für die Anweisung eingegeben. Die Abarbeitung erfolgt so, daß nach vollständiger Eingabe der Anweisung und Betätigen einer Starttaste (ET1 oder ET2) die Anweisung ausgeführt wird. Während der Abarbeitung bleibt die Eingabezeile erhalten. Nach Ausführung der Anweisung erlischt die Eingabe. Eine neue Anweisung ist möglich.

3.2.2.2. Anweisungsgruppen

Die Funktionen des PHAK werden durch folgende Anweisungsgruppen realisiert:
- Definitionsanweisungen,
- Serviceanweisungen,
- End-Anweisung.

Die Definitionsanweisungen dienen:
- der logischen Zuordnung der peripheren Geräte für das Dienstprogramm PHAK,
- der Eröffnung von Phasenbibliotheken auf KMB.

Sie sind den Serviceanweisungen zeitlich vorgelagert. Serviceanweisungen sind vom Benutzer frei wählbar. Sie umfassen folgende Anweisungen:
- Drucken von Verzeichnissen,
- Anzeigen von Verzeichnissen,
- Kopieren von Phasen auf KMB,
- Kopieren von Phasenbibliotheken auf KMB.

Die Endanweisungen dienen der Beendigung des Programms.

3.2.2.3. Definitionsanweisung

3.2.2.3.1. ASSGN-Anweisung

Mit der ASSGN-Anweisung wird die logische Gerätezuordnung für das PHAK eingestellt, d.h. es können Drucker und Floppy-Disk-Laufwerke zugewiesen werden.

Die Anweisung ASSGN P 1 , LL
spezifiziert den 1. Drucker,

ASSGN P 2 , LL
spezifiziert den Zusatzdrucker.
Die Zuweisung bleibt bis zur nächsten Anweisung ASSGN P 1 bzw. ASSGN P 2 erhalten.
Mit LL werden die Anzahl der auf den verwendeten Leporello-papier druckbaren Zeilen je Seite angegeben.

Es gilt: $6 < LL < 73$

Wird LL nicht eingegeben, so wird die Standardzeilenzahl 72 eingestellt. Vor der Druckerzuweisung muß das Leporello in Grundstellung gebracht werden.
Wurde eine dieser Anweisungen ausgeführt, so werden ab diesem Zeitpunkt alle nachfolgenden Anweisungen protokolliert. Sie ist unbedingt erforderlich, um die Funktion PRINT auszuführen.
Tritt während der Ausführung des Drucks eine Hardware-Fehler auf, so wird die Zuweisung des Druckers bzw. Zusatzdruckers aufgehoben.
Die Zuweisung wird in der Statuszeile angezeigt (vgl. Anlage 2).

Hinweis:
Diese Anweisungen sind der mit der Monitorfunktion ASN eingestellte Geräteverteilung untergeordnet.

Mit der Anweisung ASSGN 1, VOL - NAME [, Z]
wird das Floppy-Disk-Laufwerk und Datenträger zugeordnet.

Dabei bedeuten: 1 = Quelldiskette.

Über die Quelldiskette werden alle Servicefunktionen realisiert.

Mit der Anweisung ASSGN 2, VOL - NAME , Z
werden das Kassettenmagnetband-Laufwerk und der Datenträger zugerodnet.

Mit VOL-NAME wird der Datenträgername der Quelldiskette spezifiziert. Damit sind versehentliche Verwechslungen von Disketten bzw. Ändern ausgeschlossen.
Durch die Angabe Z (4-7) kann eine Zuordnung von Quelldiskette zum Floppy-Disk-Laufwerk erfolgen, d.h. dessen logischen Geräteadresse.
Fehlt diese Angabe, so erfolgt die Standard-Zuweisung:

Quelldiskette -- LW ≙ Log. Geräte-Adresse 4.

Die Laufwerksauswahl ist ebenfalls der über die Monitorfunktion ASN festgelegten Geräteverteilung untergeordnet.

Jede Anweisung bleibt solange erhalten, bis eine neue ASSGN-Anweisung erfolgt, die sich auf die Quelldiskette bezieht.
Für das Kassettenmagnetband wird mit der Anweisung ASSGN 2, ... der Kontrollname eingegeben. Dabei wird automatisch der Vermerk 'P' als 8- Buchstabe sowie der LBTYP P generiert.
Die logische Geräteadresse Z wird mit D oder E eingegeben.

Vor der Abarbeitung jeder anderen Anweisung (außer LBTYP und END) wird die durch ASSGN eingestellte Datenträgerdefinition geprüft.
Die Zuweisungen werden in der Statuszeile angezeigt.

3.2.2.3.2. LBTYP-Anweisung

Die LBTYP-Anweisung erfolgt automatisch mit ASSGN 2. Es wird immer der Bibliothekstyp

LBTYP P - Phasenbibliothek PL

zugewiesen.
Für eine Quelldiskettenfunktion z.B. Anzeige-Verzeichnis ist über die LBTYP-Anweisung die Phasenbibliothek zu definieren.

3.2.2.3.3. NEW-Anweisung

Mit der NEW-Anweisung wird ein Kassettenmagnetband für die Aufnahme von Phasen vorbereitet.
Diese Anweisung dient auch zum Löschen ungültiger Bänder.
Am Bandanfang wird ein Buch mit dem eingegebenen VOL-NAMEN (6 -Zeichen) aufgezeichnet. Dieses Buch dient zur Kontrolle und wird in der COPY-Funktion geprüft.

3.2.2.4. Serviceanweisungen

3.2.2.4.1. PRINT-Anweisung

Folgende PRINT-Anweisung ist zugelassen:

PRINT DIC

Mit der Anweisung PRINT DIC können Verzeichnisse der Phasenbibliothek gedruckt werden.

Hinweis:
Bei gelöschten Phasen wird kein Name gedruckt (kann nur bei Diskette auftreten).
Drucker, Bibliothekstyp und Quelldiskette bzw. KMB müssen vorher durch Definitionsanweisungen festgelegt worden sein.

3.2.2.4.2.
DSPLY-Anweisung

Mit der Anweisung

DSPLY DIC

wird das Verzeichnis der jeweiligen Phasenbibliothek auf dem Bildschirm angezeigt.
Ist der Füllstand des Verzeichnisses größer als der für die Anzeige verwendete Bildschirmbereich, kann durch Betätigen der Taste ET1 die Anzeige des nächsten Verzeichnisteils vorgenommen werden. Die Taste ET2 beendet die Anzeige des Verzeichnisses.
Bibliothekstyp und Quelldisketten bzw. KMB müssen definiert worden sein.

Hinweis:
Gelöschte Phasen werden als Leerzeichen im Namensfeld dargestellt (nur bei Diskette möglich).

3.2.2.4.3.
COPY-Anweisung

Das Kopieren von Phasen nach KMB wird durch die Anweisung

COPY NAME 1, NAME 2 ...

realisiert.

Vor Ausführung der Anweisung wird geprüft, ob Quelldiskette auf Laufwerk definiert wurden und auf dieser Diskette eine Bibliothek des zugewiesenen Typs P vorhanden ist.
Darüber hinaus wird geprüft, ob alle zu kopierenden Phasen in der Quellbibliothek enthalten und auf dem Band noch nicht enthalten sind.
Vor dem Kopieren wird der Füllstand des Verzeichnisses und Datenteils bezüglich der aufzunehmenden Phasen geprüft.
Lücken (gelöschte Namen) im Verzeichnis der Quelldiskette werden übergangen, Quellbücher werden verdichtet übertragen.
Der Datenträger KMB wird lückenlos mit Phasen gefüllt.

Die Anweisung COPY ,ALL

bewirkt das Kopieren der gesamten Phasenbibliothek auf das KMB in verdichteter Form.
Für den Vergleich der beiden Verzeichnisse bezüglich gleicher Namen gilt das oben gesagte.
Handelt es sich bei der Zielbibliothek um eine fortgeschriebene Bibliothek, so wird diese um den Inhalt der Quellbibliothek (verdichtet) fortgeschrieben.
Ist die Zielbibliothek neu (noch keine Phase enthalten), so stellt das Duplikat im Ergebnis der Kopierfunktion die verdichtete Quellbibliothek dar.

Hinweis:
Für das Kassettenmagnetband wird während der Programmbearbeitung ein fiktives Verzeichnis aufgebaut. Es wird nicht auf den Datenträger ausgegeben.

3.2.2.5.
END-Anweisung

Während des gesamten Programmlaufs des PHAK kann die Monitortaste zum Verlassen des Programms nicht benutzt werden. Manipulationen werden damit weitgehend ausgeschlossen.

Deshalb wird zur Beendigung des Dienstprogramms die Anweisung

END

benutzt.
Die Programmwiederholung ist mit der Monitorfunktion

NEW

möglich.

3.2.3. Fehlerbehandlung

Fehlerbehandlung und Fehlermeldungen werden durch das Dienstprogramm weitgehend selbst übernommen. Das betrifft insbesondere:

- Syntaktische Analyse der Anweisungen,
- Prüfung der Operanden auf Zulässigkeit,
- Übertragung von Floppy-Disk nach Kassettenmagnetband,
- Druckanweisungen.

Die Fehlerbehandlung erfolgt nach dem Standard.
Im Fehlerfalle wird die jeweilige Fehlerart (Fehler-Nr.) im Errorfeld (vgl. 3.2.1.3.5.) angezeigt. In der Eingabezeile wird die auszuführende Anweisung so markiert, daß der Bezug zur Fehlerursache leicht herzustellen ist.
Die Fehlerkorrektur kann auf der nachfolgenden Bildschirmzeile vorgenommen werden.

Hinweis zur Organisation der Programmerzeugung:

Für Programmkorrekturen und eventuelle Zerstörungen der gespeicherten Informationen auf der Diskette wird empfohlen, Duplikate auf verschiedenen Disketten zu führen.

4.
Ein- und Ausgabedaten

4.1.
Eingabedaten

Unter Eingabedaten werden die
- Bibliotheken, gespeichert auf Diskette, und
- Anweisungen, eingegeben über Tastatur,

verstanden.

4.2.
Ausgabedaten

Unter Ausgabedaten werden die erzeugten oder fortgeschriebenen Phasenbibliotheken verstanden.

Anwendungsbeschreibung

Programmbezeichnung: Mischprogramm
Kurzbezeichnung: MIXD 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Das Dienstprogramm ermöglicht es, zwei sortierte, homogene Dateien physisch zu einer, in der Satzfolge sortierten sequentiellen Datei zu vereinigen.
Dabei ist es unerheblich, ob sich die zu mischenden Dateien auf einer oder mehreren Disketten befinden. Zugelassen sind alle physischen Satzlängen sowie geblocktes Satzformat.

2. Anwendungsbedingungen

Die Zieldatei (Ausgabedatei) muß vor Programmbeginn initialisiert sein (HDR1 generieren). Der Datenbereich muß so definiert sein, daß er alle Sätze der zu mischenden Dateien aufnehmen kann. Es sind alle vier physischen Satzlängen zugelassen. Geblocktes Satzformat kann verarbeitet werden, Segmentierung ist nicht erlaubt. Die maximale Blocklänge ist mit 1 K Bytes festgelegt. Bei ungeblocktem Format entspricht das der maximal zulässigen logischen Satzlänge.

Die Quelldateien werden als Sortiert und sequentiell vorausgesetzt.
Die Sortierung kann dabei auf- oder absteigend sein, muß aber in beiden dieselbe sein.
Für das Sortiermerkmal sind maximal 32 Positionen innerhalb eines Satzes zugelassen. Diese müssen aber nicht notwendig zusammenhängend sein. Markierte Sätze werden wahlweise übernommen bzw. überlesen.
Als Dialoggerät dient der Bildschirm (1024 oder 1920 Byte).
Für einen sinnvollen Ablauf des Programms muß die ständige Zugriffsmöglichkeit auf die Dateien gewährleistet sein. Mehrdatenträgerbetrieb ist zugelassen, allerdings nur auf dem Laufwerk, das der Datei zugeordnet ist.

Technische Mittel

- Speicherbedarf:
  - a) Programm: 4,5 K Bytes
  - b) E/A-Bereiche: 2 K Bytes
  - c) Pufferbereiche: 3 K Bytes
  - Summe: 9,5 K Bytes
- Gerätekonfiguration:
  - . Floppy-Disk 1 - 3 Laufwerke
  - . Bildschirm (1024 oder 1920 Byte) als Dialoggerät

3. Beschreibung von Aufgabe und Lösung

Es ist Ziel, zwei bereits sortierte Dateien zu einer zu vereinigen, wobei diese dann wiederum sortiert sein soll.
Als Speichermedium wird die Diskette verwendet. Zugelassen sind alle physischen Satzlängen, geblocktes Satzformat bei einer maximalen Blocklänge von 1 K Bytes sowie Mehrdatenträgerbetrieb.

Markierte Sätze können wahlweise übernommen oder überlesen werden. Fehler bei der Eingabe bzw. Kommunikation und der Programmabarbeitung werden erkannt, die Art des Fehlers angezeigt und nach Quittung entsprechend behandelt.
Als Kommunikationsgerät dient der Bildschirm.
Die gewünschte Zieldatei ist vom Anwender auf der entsprechenden Diskette im Datenträgerkennsatz mit FGEN einzurichten.

Als Eingangsdaten bzw. zur Kommunikation werden zu Beginn folgende Daten eingelesen:

1. Log. Geräteadresse und Dateiname der 1. Quelldatei,
2. Log. Geräteadresse und Dateiname der 2. Quelldatei,
3. Sollen gelöschte Sätze übernommen werden?
   (ja = Y, nein = N)
4. Log. Geräteadresse und Dateiname der Zieldatei,
5. Sortierung aufsteigend? (ja = Ø, nein = 1),
6. Schlüsselposition dezimal (von ... bis), maximal 32 Pos.

Nach jeder Eingabe erfolgt eine Kontrolle auf Zulässigkeit. Fehler werden mit dem entsprechenden Fehlercode angezeigt, die Korrektur wird ermöglicht.
Für die logische Geräteadresse der beiden Quelldateien sowie der Zieldatei und für die Eingaben 3 und 5 bietet das Programm Standardzuweisungen an, die entweder mit ET1 quittiert oder über die Alpha-Tastatur geändert werden können.
Wird die Eingabe mit ET2 abgeschlossen, so erfolgt ein Sprung an den zentralen Haltepunkt, von dem aus entweder neu eingegeben oder in das Monitorprogramm gesprungen werden kann.

Eine Ausnahme bildet dabei Eingabe 6. Ist der Eingabeteil beendet, bietet ein Haltepunkt die Möglichkeit, die Eingaben zu kontrollieren und mit Drücken der ET1-Taste das Programm fortzusetzen bzw. mit der ET2-Taste den Eingabeteil erneut anzuspringen und eventuelle Korrekturen vorzunehmen.

Der Programmablauf kann in der letzten Bildschirmzeile visuell verfolgt werden. Nach dem Lesen des ersten Satzes der beiden Quelldateien wird anhand des Sortiermerkmals entschieden, welcher der beiden Sätze auf die Zieldatei ausgegeben wird. Je nachdem, von welcher Datei (Quelldatei) der ausgegebene Satz stammte, wird von dort auch ein neuer Satz sequentiell eingelesen.
Beim Vergleich der Schlüssel kontrolliert das Programm intern die Sortierung in den Quelldateien und weist unsortierte Dateien zurück. Sind Sortiermerkmale gleich, wird der Satz der 1. Quelldatei ausgegeben. Nach Beendigung des Mischvorganges werden alle drei Dateien geschlossen, und das Programm steht an einem Haltepunkt.
Wird innerhalb des Programms zum zentralen Haltepunkt gesprungen, bleiben die Eingaben erhalten!
Tritt bei der Abarbeitung des Programms ein Fehler auf, der zum Abbruch führt, steht das Programm am zentralen Haltepunkt, und die Dateien sind geschlossen. Das bedeutet, daß die Sätze, die bereits auf die Zieldatei ausgegeben worden sind, für den Zugriff erhalten bleiben.
Anhand der Fehlermeldung bleibt dem Anwender nun zu entscheiden, ob das Programm abgebrochen werden muß oder ein erneuter Start mit den notwendigen Änderungen erfolgen kann.

4.
Ein- und Ausgangsdaten

Eingangsdaten

- Programmparameter (über Tastatur),
- Datensätze der Quelldateien.

Programm: MIXD 1520 (SIEX)

Ausgangsdaten

- Gemischte, sortierte Zieldatei,
- Quelldateien unverändert,
- Anzahl der gelesenen Sätze von Quelldatei 1,
- Anzahl der gelesenen Sätze von Quelldatei 2,
- Anzahl der geschriebenen Sätze von Zieldatei.

Anwendungsbeschreibung

Programmbezeichnung: Kopieren von Dateien von/nach Magnetbandkassette
Kurzbezeichnung: COCA 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

COCA hat die Aufgabe, die n-te Datei (Format: SIMPLE oder BASIC oder C O M P A C T) einer Quellkassette als m-te Datei eines auszuwählenden, möglicherweise neuen Formats auf eine Zielkassette zu kopieren bzw. die m-te Datei zu erweitern.

2. Anwendungsbedingungen

Geräteausstattung:
2 Magnetbandkassettengeräte,
Bildschirm 1 oder 2 K,
≧ 6 K Arbeitsspeicher,
Dateiorganisation auf Quell- und Zielkassette nach den Festlegungen von SIOS.

3.
Beschreibung von Aufgabe und Lösung

3.1.
Aufgabe

Durch Bedienereingaben ist die zu kopierende Datei der Quellkassette auszuwählen. Der Dateityp ist vorzugeben/zu prüfen. Ebenfalls durch Bedienereingaben ist auszuwählen:

- der Dateityp der zu kopierenden Datei auf der Zielkassette; dieser Typ kann vom Typ der Quelldatei abweichen;
- der Platz der Datei auf derZielkassette,
- neue Datei / Erweitern der Zieldatei.

Das Kopieren soll Satz für Satz erfolgen. Fehler sind nach dem Standard zu behandeln.

3.2.
Lösung

Das Programm besteht aus den Teilen:

- Kommunikation zur Quelldatei,
- Kommunikation zur Zieldatei,
- Kopiervorgang,
- Fehlerbehandlung.

4.
Eingangsdaten / Ausgangsdaten

4.1.
Eingangsdaten

- Zu kopierende Datei (Quelldatei),
- Kommunikationseingaben.

4.2.
Ausgangsdaten

- Kopierte Datei (Neue Datei oder erweiterte Datei),
- Fehlermeldungen.

Anwendungsbeschreibung

Programmbezeichnung: Physisches Kopieren von Disketten
Kurzbezeichnung: CODP 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Das Programm dient zum physischen Kopieren von 8"- und 5"-Disketten, wobei auch nur einzelne Spuren kopiert werden können.
Es können Disketten folgender Formate kopiert werden:

a) 128 Byte/Sektor,
b) 256 Byte/Sektor,
c) 512 Byte/Sektor,
d) 1024 Byte/Sektor.

Die Spur Ø muß jedoch immer das Format 128 Byte/Sektor besitzen.
Quell- und Zieldiskette müssen das gleiche Format besitzen.
Kopieren von 5"- auf 8"-Diskette und umgekehrt ist möglich.

2.
Anwendungsbedingungen

Voraussetzung sind Disketten gleicher Sektorlänge.
Auftretende Fehler werden protokolliert.

Konfiguration: 1 Bildschirm mit 1024 oder 1920 Zeichen,
1 Drucker (nur bei Variante mit Fehlerdruck!),
2 Floppy-Disk-Laufwerke.

Speicherbedarf: = 6 K Byte

3.
Beschreibung von Aufgabe und Lösung

Als Dialoggerät dient der Bildschirm.

1. Diskettenzuweisung über Bildschirm (Anlage 1) mit Laufwerk, Name, sowie von welcher Spur - bis zu welcher Spur kopiert werden soll.
   Standard ist die volle Diskette: Spur ØØ - 74,
   bzw. bei 5"-Diskette: Spur ØØ - 37.
2. Festlegen des COPY-Ablaufes bei Auftreten von Fehlern
   a) ohne Stopp,
   b) mit Stopp,
   c) Protokollieren von Fehlern.

a) Kopieren bei evtl. Fehler (ohne Stopp)
   - Schreibfehler: Abbruch COPY (Neubeginn möglich)
   - Lesefehler: 3 Varianten für Sektorbelegung:
     . Ausgabe fehlerhafter Sektor unverändert: (VOL 1),
     . Ausgabe fehlerhafter Sektor mit 'ØØ'H aufgefüllt,

. Ausgabe fehlerhafter Sektor mit 'FF'H aufgefüllt.

b) Kopieren bei evtl. Fehler --- Stopp
- dann erst Entscheidung über weiteren Weg (wie unter a).

c) Darüber hinaus ist die Ausgabe einer Fehlerliste über Drucker in allen 3 Fällen möglich, wobei bei "Nicht-Druck" die Lesefehler gezählt werden (Anzeige auf Bildschirm).
In der Statuszeile erfolgt die Anzeige:
- VOL-NAME, ØØ - 74
- COPY - Art
- LW1 } Spurzähler, der den Bearbeitungsstand anzeigt
- LW2 }
(s. Anlage 2)

4. Eingangsdaten und Ergebnisse

a) Eingangsdaten: . 8"- oder 5"-Quelldiskette, die kopiert werden soll
. 8"- oder 5"-Zieldiskette

b) Ausgangsdaten: - Kopierte Zieldiskette

Anlage 1

CODP 1520 (SIEX) V Ø.3

VOL 1 - NAME: 4,....., ØØ - 74

STATEMENTS

VOL 1 - NAME:	n, NAME, ØØ - 74	COPY (VOL 1):	Y/N
VOL 2 - NAME:	n, NAME, ØØ - 74	COPY ØØ:	Y/N
COPY STOP :	Y/N	COPY FF:	Y/N
END :	Y/N	PRINT :	Y/N

Anlage 2

COPY=Art==X,NAME1=ØØ-74=Sp.===>==Y,NAME2=ØØ-74=Sp.=COPY a)==

a) Variante der Ausgabe fehlerhafter Sektor

Anwendungsbeschreibung

Programmbezeichnung: Textverarbeitung
Kurzbeschreibung: TEXT 1520 (SIBX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1. Zweckbestimmung des Programms

Das Programm dient der Eingabe von Texten beliebiger Struktur, ihrer Änderung, Speicherung auf Floppy-Disk sowie dem Druck von Texten auf Leporello/Journal (60 Zeilen à 60 Zeichen).

2. Anwendungsbedingungen

Voraussetzung sind Disketten mit einer Sektorlänge von 128 Byte. Die Disketten müssen mit einem Dateikennsatz (HDR) mit folgendem Aufbau versehen sein:

POS 6	Dateiname	TEXT
POS 23	Blocklänge	00128
POS 28	Satzmerkmal	B
POS 29	BOE	xxxxx (01001)
POS 35	EOE	xxxxx (beliebig)
POS 44	Datenaustauschniveau	E
POS 54	Satzlänge	0064
POS 75	EOD	xxxxx (01001)

Anwendungsbeschreibung

Programm: TEXT 1520 (SIEX)

Mit diesem Programm können nur Texte verarbeitet werden, die auf MFG/GBG hergestellt wurden. Für den Druck der Texte ist das Format A4 vorgesehen. Es kann auf Leporello bzw. Journal (linker Walzenteil) gedruckt werden.

Technische Voraussetzungen

Konfiguration: 1 Bildschirm mit 1024 oder 1920 Zeichen,
1 Drucker,
1 Floppy-Disk-Laufwerk
(2 Floppy-Disk-Laufwerke bei Reorganisation).

Speicherbedarf: 8 K Byte

3.
Beschreibung von Aufgabe und Lösung

Beschreibung der Aufgabe

Das Programm gliedert sich in folgende einzelne Programmteile:

Kommando 1 - Schreiben Text,
Kommando 2 - Drucken Text,
Kommando 3 - Drucken Verzeichnis,
Kommando 4 - Cursor-Korrektur,
Kommando 5 - Erweitern,
Kommando 6 - Löschen Zeile,
Kommando 7 - Löschen Buch,
Kommando 8 - Anzeigen Text,
Kommando 9 - Reorganisation,
Kommando 10 - Abschluß des Programms ⟶ MONITOR.

Als Dialoggerät soll der Bildschirm dienen.
Bedienungshinweise zur Auswahl der Programmteile und der Abarbeitung des Programms werden auf dem oberen Bildschirmteil angezeigt.

Die Speicherung der Datensätze (Zeilen) auf Diskette erfolgt geblockt (2 log. Sätze/Sektor). Die Sektorgröße beträgt 128 Byte. Mit dieser Blockung ist es möglich, ungefähr 3700 Zeilen/Diskette zu speichern.

Beschreibung der Lösung

Das Programm wird über Monitor geladen und gestartet. Nach dem Start erscheinen die Programmauswahl und die dazugehörende Bedienungsanweisung auf dem Bildschirm.
Als erste Eingabe wird der Name des zu bearbeitenden Textes verlangt und in das Titelverzeichnis auf der Diskette eingetragen. Anschließend wird die Verzweigung zu den einzelnen Programmteilen über eine Kommandoeingabe realisiert.

- Schreiben Text
- Eingabe: Textzeile

Der eingegebene Text wird sofort auf dem Bildschirm angezeigt. Nachdem der Bildschirm gefüllt ist (BAB 1 = 16 Zeilen, BAB 2 = 24 Zeilen), erfolgt die Ausgabe der ersten Zeile des Bildschirms auf Floppy-Disk.
Der gesamte Bildschirminhalt wird anschließend um eine Zeile nach oben verschoben und die letzte Bildschirmzeile gelöscht. Der weitere Text wird dann ständig auf die letzte Bildschirmzeile geschrieben und die erste Zeile jeweils auf Floppy-Disk ausgegeben.
Zur Kontrolle der geschriebenen Zeilen erscheint am linken Rand die Zeilenanzahl.
Eine Korrektur innerhalb einer Zeile ist über die Rückschritttaste realisierbar. Über die C-Taste ist eine Wiederholung der gesamten Zeile möglich.

Selektorbedingungen

ET 2 - Korrektur der vorhergehenden Zeile,
SEL 2 - Briefkopf,
SEL 3 - Ende Seite (Seite vorzeitig beenden),
SEL 4 - Textende,
SEP 9 - Abbruch.

- Drucken Text

- Eingabe: Anzahl Druckwiederholungen

Der entsprechende Text wird auf Journal/Leporello (max. 60 Zeichen/Zeile, 60 Zeilen/Seite) ausgedruckt. Der Seitenwechsel erfolgt dabei automatisch über Zeilenzähler bzw. über ein Steuerzeichen, das im Datensatz enthalten sein kann. Der Druck des Textes mit der "aktuellen Zeilen-Nr." ist über Selektorentscheidung möglich.

- Drucken Verzeichnis

Mit diesem Programmteil kann das Titelverzeichnis

Satz-Nr. Textname

auf Leporello/Journal ausgedruckt werden.

- Cursor-Korrektur

- Eingabe: Zeilen-Nr. der Zeile, die korrigiert werden soll

Anhand der Zeilen-Nr. wird der entsprechende Datensatz gelesen und auf dem Bildschirm angezeigt. Bei richtiger Anzeige kann diese Zeile geändert bzw. neu geschrieben werden. Anschließend wird diese Zeile wieder auf Diskette ausgegeben.

- Erweitern
- Eingabe: Zeilen-Nr. der Zeile, an die an- bzw. eingefügt wird

Über die eingegebene Zeilen-Nr. wird der entsprechende Datensatz gelesen und zur Kontrolle auf dem Bildschirm angezeigt. Bei richtiger Eingabe der Zeilen-Nr. kann anschließend mit dem Schreiben der Ergänzung begonnen werden. Die geschriebenen Zeilen werden wieder auf dem Bildschirm angezeigt (analog "Schreiben Text") und auf die nächsten freien Speicherplätze auf Diskette ausgegeben.
Nach Abschluß der Ergänzung wird in dem Datensatz, an den an- bzw. eingefügt wurde, die Satz-Nr. des ersten eingefügten Datensatzes eingetragen.
Der letzte Datensatz der Ergänzung erhält dann den Rückverweis auf den normalen Textablauf.

2	3 -→ y	4	.....	x	3
1. DS	2. DS	3. DS		y. DS	x. DS

- Löschen Zeile
- Eingabe: Zeilen-Nr. der zu löschenden Zeile

Anhand der eingegebenen Zeilen-Nr. wird die Zeile gelesen und zur Kontrolle auf dem Bildschirm angezeigt. Bei richtiger Eingabe der Zeilen-Nr. wird diese Zeile (Datensatz) - mit "L" gekennzeichnet - wieder ausgegeben.

- Löschen Buch

Der zu löschende Textname wird zur Kontrolle auf dem Bildschirm angezeigt und anschließend im Titelverzeichnis der Diskette gelöscht.

- Anzeigen Text
- Eingabe: Zeilen-Nr.
  (Anzeige ab Beginn: nur ET1)

Der entsprechende Text wird ab Beginn bzw. ab dieser eingegebenen Zeilen-Nr. auf dem Bildschirm angezeigt. Mit der Textzeile wird gleichzeitig die Zeilen-Nr. (Satz-Nr.) angezeigt.

Diese Zeilen-Nr. ist Voraussetzung für die Programmteile

- Cursor-Korrektur,
- Erweitern,
- Löschen Zeile und
- Anzeigen Text ab Zeilen-Nr.

Ein Abbruch der Anzeige ist an jeder beliebigen Stelle möglich.

- Reorganisation (Voraussetzung: 2. Laufwerk)

In diesem Programmteil werden auf einer neuen Diskette, die mit dem entsprechenden Dateikennsatz versehen sein muß, die bisher geschriebenen Texte verdichtet.
Gelöschte Titel und der dazugehörende Text werden nicht mit übernommen. Die Lücken von gelöschten Sätzen fallen ebenfalls weg.

4.
Ein- und Ausgabedaten

Eingabedaten

- Text,
- Texte, die auf Diskette gespeichert vorliegen,
- Anweisungen, die über Tastatur eingegeben werden.

Ausgabedaten

Ausgabedaten sind die auf Diskette gespeicherten Textkonserven bzw. der gedruckte Text im Format A4.

Anwendungsbeschreibung

Programmbezeichnung: Anzeige Inhalt Datei/Diskette
Kurzbezeichnung: DIDI 1520 (SIEX) V 1.Ø

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Mit dem Programm DIDI kann der Inhalt einer geblockten oder ungeblockten Datei satzweise (entsprechend log. Satzlänge) oder sektorweise (entsprechend phys. Sektorlänge) auf Bildschirm angezeigt werden.

2. Anwendungsbedingungen

Voraussetzung sind Disketten mit Dateien, die die Sektorlänge 128, 256, 512 oder 1024 Byte haben und der Entwicklungsrichtlinie des Kombinates für die Datenträgerkompatibilität 8"-Standarddiskette KROS 5108/01 - entsprechen.
Alle zu einer Datei gehörenden Disketten müssen sich im Zugriff befinden (kein Ein-Laufwerk-Betrieb).

Anlagenkonfiguration:

- ≥ 12 K Speicher,
- ≥ 1 Floppy-Laufwerk,
- 1 Bildschirm mit 1024 oder 1920 Byte.

3.
Beschreibung von Aufgabe und Lösung

Das Programm ermöglicht folgende Funktionen:

- Anzeige des Dateikennsatzes (HDR 1), bei Mehrdatenträgerdateien Anzeige des Dateikennsatzes der letzten Diskette der Datei,
- Anzeige eines ausgewählten Satzes der Datei hexadezimal und/oder alphanumerisch,
- Anzeige des letzten Satzes der Datei,
- Anzeige geblockter oder ungeblockter Dateien,
  Als gelöscht markierte Sätze werden nicht angezeigt.

Nach Programmaufruf und -start erscheint auf dem Bildschirm das Start-Menü.
Der Anwender gibt ein:

1.	Log. Geräteadresse	Adresse der(s) Geräte(s), in die/das Diskette(n) eingelegt wurden (4,...,7)
2.	Anzeigeform	Alphanumerisch (A) oder hexadezimal und alphanumerisch (H)
3.	Dateiname	Nach Eingabe des Dateinamens wird der Dateikennsatz angezeigt, bei Mehrdatenträgerdateien der Dateikennsatz des letzten Datenträgers.
4.	Satznummer	- Bei schlüsselindiziertem Zugriff muß diese Eingabe mit Leerstart übergangen werden. - Satznr. 99999 ermöglicht Anzeige des letzten Satzes. - Bei geblockten Dateien wird nach Eingabe der Satznr. mit S1-Taste der vollständige Sektor angezeigt, in dem sich der gesuchte Satz befindet.

5.	Schlüsselposition	1-9999 Position innerhalb des Satzes, an der Schlüssel beginnt
6.	Schlüssel	≤ 32 Stellen, Anzeige 2zeilig à 16 Stellen

Auf der 15. bzw. 23 Zeile erscheinen nach erfolgten Eingaben im Menü und mit Anzeige des ersten gewünschten Satzes:
Dateiname
- Aktuelle Satznr.,
- Schlüsselposition und Schlüssel bei schlüsselindiziertem Zugriff,
- M bei Lesen eines markierten Satzes.

Nach Anzeige der ersten Seite eines Satzes stoppt das Programm an einer Eingabe.
Nach Leerstart + wird die nächste Seite (bei Format > 128) bzw. der nächste Satz angezeigt, nach Leerstart - der vorhergehende Satz (bei allen Formaten).
Nach Eingabe einer max. 4stelligen Nummer wird die Satznummer um diese Zahl erhöht und der entsprechende Satz angezeigt. Soll ein bestimmter Satz gelesen werden, muß die entsprechende Satznummer eingegeben werden (bei Satznr. < 5 Stellen mit Vornullen eintasten).
Mit ET2 Rückkehr zum Start-Menü.

Fehlermeldungen

Die entsprechenden Ausschriften erscheinen auf der sogenannten Fehlerzeile gemäß der Richtlinie zur Fehlerbehandlung bei Fehlern während Abarbeitung des Menüs oder auf der 16. bzw. 24. Bildschirmzeile bei Fehlern während der Anzeige.

4. Ein- und Ausgangsdaten

Datensätze der entsprechenden Datei

Anwendungsbeschreibung

Programmbezeichnung: Druck Inhalt Datei / Kassette
Kurzbezeichnung: PRKA 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1.
Zweckbestimmung des Programms

Mit diesem Programm kann der Inhalt einer Datei vollständig oder satzweise ausgedruckt werden.

2.
Anwendungsbedingungen

Voraussetzungen sind Kassetten mit SIMPLE-, BASIC- oder COMPACT-Dateien.

Konfiguration: 1 Bildschirm mit 1024 oder 1920 Byte
1 Drucker
1 Kassettenlaufwerk

3.
Beschreibung von Aufgabe und Lösung

Das Programm ermöglicht folgende Funktionen:

- Druck einer COMPACT-Datei satzweise oder vollständig, alphanumerisch oder hexadezimal,
- Druck einer BASIC-Datei satzweise oder vollständig, alphanumerisch oder hexadezimal,
- Druck einer SIMPLE-Datei satzweise oder vollständig, hexadezimal oder alphanumerisch.
- Der Druck erfolgt formatiert bezogen auf die möglichen Papierbreiten von 150 mm, 210 mm, 300 mm, 460 mm.

Das Programm wird über den Monitor geladen und gestartet. Nach dem Start erscheint auf dem Bildschirm das Arbeits-Menü.

Der Anwender gibt das Laufwerk, die Druckform, den Dateinamen bzw. die Organisationsform, die Datei-Nummer, die Satznummern und die Druckbreite an.
Nach der Bedienerquittung erfolgt der Druck auf die vorgegebene Papierbreite.

4.
Ein- und Ausgangsdaten

Eingangsdaten

Eingaben mit Tastatur,
Dateien, die auf Kassetten gespeichert vorliegen.

Ausgangsdaten

Gedruckte Sätze bzw. Dateien.

Anwendungsbeschreibung

Programmbezeichnung: Erzeugen von Testdateien auf Magnetbandkassette

Kurzbezeichnung: TFGC 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

- Erzeugen von Testdateien auf Magnetbandkassette,
- Druck der Parameter der generierten Dateien (wahlweise),

2. Anwendungsbedingungen

Das Programm ist ein Hilfsprogramm zur Generierung von Dateien auf Magnetbandkassette.
Dabei sind Satzlängen bis 256 Byte zugelassen.

Es sind die Dateitypen

SIMPLE
BASIC
COMPACT

erlaubt.

3.
Beschreibung von Aufgabe und Lösung

Es sollen Dateien erzeugt werden, die sich von Satz zu Satz unterscheiden. Dazu werden einzelne POS des Datensatzes arithmetisch miteinander verknüpft oder unter Verwendung von Rechenkonstanten zyklisch verändert.
Die so generierten Dateien eignen sich zum Test einer Vielzahl von Programmen, die als Eingangsdaten eine oder mehrere Dateien benötigen (z.B. Listgenerator).
Dazu werden durch das Programm 6 Anfangswerte (IP1 ... 6) durch Eingaben spezifiziert und in die Berechnung einbezogen.
Es entstehen Datensätze, die folgenden Aufbau haben:

POS	Inhalt	
1 - 4	Lfd. Satznummer	
5 - 9	IP1 + 7 je Satz	
10 - 27	'TEST LISTGENERATOR'	Konstanter Text
28 - 32	'ØØØØ1' Konstante + 1 nach 20 Sätzen	
33 - 4Ø	Dateiname	
31 - 48	IP2 + 1 nach 27 Sätzen	
49 - 58	IP1 Konstante POS 28 ... 32	
59 - 68	'ALPHATEXT1'	Konstanter Text
69 - 78	'ALPHATEXT2'	Konstanter Text
79 - 82	Stellen 2 ... 5 von IP1	
83 - 89	IP3 + 3 je Satz	
90 - 105	Konstante Pos 28 ... 32 + Pos 49 ... 58	
106 - 115	'1234567890'	Konstante
116 - 118	'111'	Konstante
119 - 128	'ALPHATEXT3'	Konstanter Text
129 - 135	IP4 + 6 je Satz	

POS	Inhalt
136 - 147	' 12345ØØ' Konstante + 2 nach 3 Sätzen
148 - 163	IP5 + 5 - Pos 28 - 32 je Satz
164 - 175	'1Ø2Ø3Ø4Ø1Ø11' Konstante
176 - 190	(Pos 148 ... 168 + Pos 83 ... 89) x 3 je Satz
191 - 195	'125ØØ' Konstante
196 - 210	IP6 + 3 nach 40 Sätzen
211 - 225	'ØØØØ1ØØØØ1ØØØØ' Konstante
226 - 235	'ALPHATEXT4' Konstanter Text
236 - 244	'112233445' Konstante + 1 nach 12 Sätzen
245 - 255	'9ØØ8ØØ7ØØ6Ø' Konstante
256	'E' Konstanter Text

Wird eine Satzlänge < 256 spezifiziert, wird nur bis zur entsprechenden Pos. aufgezeichnet.

4. Ein- und Ausgangsdaten

Eingangsdaten

- Magnetbandkassette,
- Dateiname/Dateityp,
- Satzlänge,
- Anzahl der zu generierenden Sätze (max. 9999),
- 6 Anfangswerte IP1 ... IP6.

Ausgangsdaten

- Generierte Datei(en),
- Druck der Parameter.

Anwendungsbeschreibung

Programmbezeichnung: Erzeugen von Testdateien auf Diskette
Kurzbezeichnung: TFLI 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

- Erzeugen von Testdateien auf Diskette,
- Druck der Sätze der generierten Dateien (wahlweise),
- Druck der Parameter der Dateien (wahlweise).

2. Anwendungsbedingungen

Das Programm ist ein Hilfsprogramm zur Generierung von Dateien auf Diskette.
Voraussetzung ist, daß die Dateikennsätze (HDR) vorher z.B. unter Verwendung des Programms FGEN in der Indexspur erzeugt wurden.
Dabei sind zugelassen:

- physische Satzlänge SS 128, 256, 512 oder 1Ø24 Byte (= Sektorgröße),
- logische Satzlänge SL bis 256 Byte,
- Blocklänge BL bis 1Ø24 Byte,
- geblockte oder ungeblockte Arbeitsweise,
- gekettete Sätze.

3.
Beschreibung von Aufgabe und Lösung

Es sollen Dateien erzeugt werden, die sich von Satz zu Satz unterscheiden. Dazu werden einzelne POS des Datensatzes arithmetisch miteinander verknüpft oder unter Verwendung von Rechenkonstanten zyklisch verändert. Die so generierten Dateien eignen sich zum Test vieler Programme, die als Eingangsdaten eine oder mehrere Dateien benötigen (z.B. Listgenerator).

Dazu werden durch das Programm 6 Anfangswerte (IP1 ... 6) durch Eingaben spezifiziert und in die Berechnung einbezogen. Es entstehen Datensätze, die folgenden Aufbau haben:

POS	Inhalt	
1 - 4	Lfd. Satznummer	
5 - 9	IP1 + 7 je Satz	
10 - 27	'TEST LISTGENERATOR'	Konstanter Text
28 - 32	'ØØØØ1' Konstante + 1 nach 20 Sätzen	
33 - 4Ø	Dateiname	
41 - 48	IP2 +1 nach 27 Sätzen	
49 - 58	IP1 Konstante POS 28 ... 32	
59 - 68	'ALPHATEXT1'	Konstanter Text
69 - 78	'ALPHATEXT2'	Konstanter Text
79 - 82	Stellen 2 ... 5 von IP1	
83 - 89	IP3 + 3 je Satz	
90 - 105	Konstante Pos 28 ... 32 + Pos 49 ... 58	
106 - 115	'1234567890'	Konstante
116 - 118	'111'	Konstante
119 - 128	'ALPHATEXT3'	Konstanter Text
129 - 135	IP4 + 6 je Satz	
136 - 147	' 12345ØØ' Konstante + 2 nach 3 Sätzen	

POS	Inhalt	
148 - 163	IP5 + 5 - Pos 28 ... 32 je Satz	
164 - 175	'1Ø2Ø3Ø4Ø1Ø11'	Konstante
176 - 190	(Pos 148 ... 168 + Pos 83 ... 89) x 3 je Satz	
191 - 195	'125ØØ'	Konstante
196 - 210	IP6 + 3 nach 40 Sätzen	
211 - 225	'ØØØØ1ØØØØ1ØØØØ'	Konstante
226 - 235	'ALPHATEXT4'	Konstanter Text
236 - 244	'112233445' Konstante + 1 nach 12 Sätzen	
245 - 255	'9ØØ8ØØ7ØØ6Ø'	Konstante
256	'E'	Konstanter Text

Wird eine Satzlänge < 256 spezifiziert, wird nur bis zur entsprechenden Pos. aufgezeichnet.

4. Ein- und Ausgangsdaten

Eingangsdaten

- Diskette mit Dateikennsätzen (HDR),
- Dateiname,
- Satzlänge als i•SS,
- Länge Blockpuffer als i•SS
  (bei ungeblockten Dateien = Ø),
- Anzahl der zu generierenden Sätze (max. 9999),
- 6 Anfangswerte IP1 ... IP6.

Ausgangsdaten

- Generierte Datei(en),
- Druckliste der generierten Sätze der Datei (en),
- Druck der Parameter.

Anwendungsbeschreibung

Programmbezeichnung: Druck Datei / Diskette
Kurzbezeichnung: PRDI 1520 (SIEX) V 1.0

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgangsdaten

1. Zweckbestimmung des Programms

Mit dem Programm PRDI kann der Inhalt einer geblockten oder ungeblockten Datei satzweise oder vollständig ausgedruckt werden.

2. Anwendungsbedingungen

Voraussetzung sind Disketten mit Dateien, die die Sektorlänge 128, 256, 512 oder 1024 Byte haben und der Entwicklungsrichtlinie des Kombinats für die Datenträgerkompatibilität 8"-Standarddiskette KROS 5108/01 entsprechen. Alle zu einer Datei gehörenden Disketten müssen sich im Zugriff befinden (kein Ein-Laufwerk-Betrieb).

Anlagenkonfiguration:

- ≥ 12 K Speicher,
- ≥ 1 Floppy-Laufwerk,
- 1 Bildschirm mit 1024 oder 1920 Byte,
- 1 Drucker.

3.
Beschreibung von Aufgabe und Lösung

Das Programm ermöglicht folgende Funktionen:

- Druck des Dateikennsatzes (HDR 1), bei Mehrdatenträgerdateien Druck des Dateikennsatzes der letzten Diskette der Datei,
- Druck eines ausgewählten Satzes der Datei hexadezimal oder alphanumerisch,
- Druck mehrerer Sätze der Datei hexadezimal oder alphanumerisch,
- Druck einer vollständigen Datei hexadezimal oder alphanumerisch,
- Der Druck erfolgt formatiert, bezogen auf die möglichen Papierbreiten von 150 mm, 210 mm, 300 mm, 460 mm.
- Als gelöscht markierte Sätze werden nicht gedruckt.

Nach Programmaufruf und -start erscheint auf dem Bildschirm das Start-Menü.

Der Anwender gibt ein:

1.	Phys. Geräteadresse(n)	4 ... 7
2.	Anzeigeform	A - alphanumerisch H - hexadezimal
3.	Dateiname	- 8 Stellen (1. Zeichen ≠ Ziffer)
4.	Satznummer	00001 ... 65533 5stellig 99999 = letzter Satz der Datei von: 00000 bis: 00000 = vollst. Datei
5.	Schlüsselposition	1-9999 Position innerhalb des Satzes, an der Schlüssel beginnt.
6.	Schlüssel	≤ 32 Stellen

Bei Falscheingabe kann mit Kursor bzw. ET2-Taste (Sprung zur vorhergehenden Eingabe) korrigiert werden.
Vor dem Druck des gewünschten Satzes (oder der gewünschten Sätze) wird der Dateikennsatz, die akt. Satznummer, die Schlüsselposition mit Schlüssel bei schlüsselindiziertem Zugriff und M bei Lesen eines markierten Satzes gedruckt.
Mit Bedienen der ET2-Taste kann die vorhergehende Eingabe wiederholt werden.
Mit A ET2 Rückkehr zum Start-Menü.

Fehlermeldungen

Die entsprechenden Ausschriften erscheinen auf der sogenannten Fehlerzeile gemäß der Richtlinie zur Fehlerbehandlung.

4. Ein- und Ausgangsdaten

Datensätze der entsprechenden Datei.

Anwendungsbeschreibung

Programmbezeichnung: Emulation Abonnentenpunkt 62 (EC 8562)
Kurzbezeichnung: EM62 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung

1. Zweckbestimmung des Programms

Das Programm EM62 hat die Aufgabe, das Anschluß- und Geräteverhalten der im Datenfernverarbeitungssystem mit START-STOP-Prozedur (asynchrone Prozedur) arbeitenden Bildschirmgeräte EC 8562/EC 8564 so nachzubilden, daß die robotron-Bürocomputer A 5120/A 5130 in der Version als Terminal K 8931 (EC 8565/8577) in den Zugriffsmethoden des ESER-Betriebssystems wie das Bildschirmsystem EC 8562/EC 8564 unterstützt werden und damit die Dialogarbeit mit ESER-Rechnern realisiert wird.

2. Anwendungsbedingungen

Erforderliche Geräteausstattung

- Speicherbedarf: $\geq$ 15 K Byte Arbeitsspeicher (einschl. Zwischenspeicher)
- Externe Geräte: Bildschirm 16 x 64 Zeichen, 24 x 80 Zeichen (davon 12 x 80 genutzt)

≧ 1 FD- oder MBK-Laufwerk (zum Programmladen),
Wahlweise Drucker (log. ADR ØB),
Asynchroner DFÜ-Anschluß (IFSS bzw. V 24)

Vor dem Starten des Emulationsprogramms müssen die Übertragungsstrecken zugeschaltet sowie ggf. MODEM und MPD eingestellt sein auf:

- Halbduplex,
- Zweidraht,
- Gleiche Übertragungsgeschwindigkeit wie Bürocomputer (durch Techniker einzustellen, am Terminal schaltbar!).

3. Beschreibung von Aufgabe und Lösung

Durch das Programm EM62 wird das Anschluß- und Geräteverhalten der mit START-STOP-Übertragung arbeitenden Bürocomputer EC 8565 bzw. EC 8577 identisch gestaltet dem der in Datenfernverarbeitungssystemen des VEB Kombinat Robotron eingesetzten Bildschirmgeräte EC 8562 bzw. EC 8564 (aus der UVR).
Damit ist die Unterstützung der Bürocomputer in den Zugriffsmethoden der ESER-Betriebssysteme abgesichert. Hinsichtlich anderer Betriebssysteme bestehen günstige Koppelbedingungen.

3.1. Anschlußverhalten

Unter "Anschlußverhalten" werden alle Faktoren zusammengefaßt, die zur Realisierung des Datenaustausches zwischen dem zu emulierenden Gerät und dem Komplex der EDVA eingehalten werden müssen.

Dazu gehören:
- die Steuerung der Schnittstelle,
- die Übertragungsprozedur.

3.1.1. Steuerung der Schnittstelle

Die Steuerung der Schnittstelle (12 bzw. V 24) erfolgt in den Bürocomputern durch die Komponenten des asynchronen DFÜ-Anschlusses.
Bei der Steuerung der Schnittstelle durch den übergeordneten Rechner müssen die technischen Parameter dieser Komponenten beachtet werden, die im Systemhandbuch (Teil I, Pkt. 6) aufgeführt sind.
Zu beachten ist, daß nur Zweidrahtleitungen als Datenübertragungskanäle einsetzbar sind und deshalb infolge der notwendigen Umschaltung der MODEMS die physikalisch bedingte Umschaltverzögerung berücksichtigt werden muß.

3.1.2. Übertragungsprozedur

Die Übertragungsprozedur wird zum Teil durch die Komponenten des asynchronen DFÜ-Anschlusses der Bürocomputer (Betriebssystem) entsprechend Systemhandbuch (Teil I, Pkt. 6), zum Teil vom Emulationsprogramm realisiert.
Von den Kommandofolgen, die zur Einleitung des Datenaustausches notwendig sind (Selekt- bzw. Pollfolgen[1]), werden ADR 2 (Geräteadresse) und Kommandocode und nach erfolgtem Nachrichtenempfang die Nachricht von den Komponenten des DFÜ-Anschlusses an das Emulationsprogramm zur Auswertung (entsprechend Geräteverhalten und Kommandovorrat des zu emulierenden Gerätes EC 8562/EC 8564) übergeben.

Vom Emulationsprogramm wird den Komponenten des DFÜ-Anschlusses umgekehrt die eingegebene bzw. zu sendende Nachricht übergeben.

[1]) Selektfolge ≙ Empfangskommando-Folge für das Terminal
Pollfolge ≙ Sendekommando-Folge für das Terminal

3.1.2.1. Auswertung der Kommandofolgen

Entsprechend dem Kommandovorrat (s. Anlage) ermittelt das Emulationsprogramm bei Empfangskommandos:
- aus der übergebenen Geräteadresse, ob die nachfolgende Nachricht auf Drucker oder Display,
- aus dem Kommando-Code, ab welcher Bildschirmposition sie auszugeben ist und welches Format die Nachricht hat.

Bei Sendekommandos ermittelt das Emulationsprogramm
- aus der Geräteadresse, ob Drucker oder Display abgefragt werden,
- aus dem Kommandocode, ab welcher Bildschirmposition auszugeben ist und welches Format die Nachricht haben soll.

Die Nachricht wird dann in der geforderten Form an die Komponenten des DFÜ-Anschlusses übergeben.

3.1.2.2. Nachrichtenformat

Ausgabe- (Empfangs-) Nachricht:

STX TEXT ETB/ETX LRC bei Kommando H'40', H'60', H'70'
STX ADR3 TEXT ETB/ETX LRC bei Kommando H'50'
STX ADR3 ADR4 TEXT ETB/ETX LRC bei Kommando H'58'

Eingabe- (Sende-) Nachricht:

STX ADR2 TEXT ETB/ETX LRC bei Kommando H'20', H'30'
STX ADR2 ADR3 ADR4 TEXT ETB/ETX LRC bei KDO H'28'

ADR2 = Geräteadresse
ADR3 = Zeilenadresse
ADR4 = Spaltenadresse

3.2. Geräteverhalten

Unter Geräteverhalten werden alle Faktoren zusammengefaßt, die beachtet werden müssen, um ein kompatibles Verhalten der Geräte hinsichtlich der Kommunikationseigenschaften zwischen Gerät und Bediener einerseits und der Datenaustauschprozedur andererseits zu gewährleisten.

Das umfaßt:
- Sonderverwendung von Codezeichen,
- Auswahl, Codierung und Wirkung der Steuerzeichen,
- Organisation der Bildschirmarbeit.

3.2.1. Sonderverwendung von Codezeichen

Aus dem KOI-7-Code (s. Systemhandbuch, Teil I, Anlage 3) werden die Zeichen ! (H'21') und # (H'23') als Ende- bzw. Startzeichen verwendet.
Diese Zeichen können deshalb nicht als Textzeichen verwendet werden.

In den Bürocomputern werden die Codezeichen
H'1Ø', H'18', H'1C' zur Kennzeichnung geschützter Bereiche bzw. Positionen genutzt und dürfen deshalb von der EDVA nicht gesendet werden.

3.2.2. Steuerzeichen

3.2.2.1. Übertragungssteuerzeichen

Die Übertragungssteuerzeichen

STX, ETX, EOT, ACK, NAK, ETB

werden nur im Niveau der Übertragungsprozedur durch die Komponenten des DFÜ-Anschlusses verwendet.
Sie belegen keine Speicherpositionen und werden dem Emulationsprogramm nicht übergeben.

3.2.2.2. Gerätesteuerzeichen

3.2.2.2.1. Formatsteuerzeichen

Die Formatsteuerzeichen HT, LF, VT und CR können nur von der EDVA gesendet werden.
Sie werden mit dem Empfangstext an das Emulationsprogramm übergeben und von diesem entsprechend ausgewertet.
Vom Bediener des Bürocomputers können diese Zeichen nicht erzeugt werden (Abweichung gegenüber EC 8562)!
Die Zeichen werden nicht angezeigt und belegen damit keine Position im Bildschirm (Abweichung gegenüber EC 8562)!

Das Emulationsprogramm wertet die Formatsteuerzeichen so aus, daß sie unter Beachtung der geschützten Bereiche (s. Pkt. 3.2.3.) eine Cursorpositionierung auf dem Bildschirm bzw. bei Ausgaben auf dem Drucker eine Positionierung des Druckkopfes bzw. Formulars bewirken.

3.2.2.2.2. Registerumschaltzeichen SO und SI

Der Zeichenvorrat der Bürocomputer umfaßt entweder
- große und kleine lateinische Zeichen,
- große und kleine kyrillische Zeichen,
- große lateinische und große kyrillische Zeichen.

Damit sind alle Codepositionen eindeutig. Die Registerumschaltzeichen SO und SI sind für die Bürocomputer ohne Bedeutung.
Das Emulationsprogramm blendet von der EDVA übertragene Zeichen SO bzw. SI aus dem Empfangstext aus.
Die Ausgabe im Bürocomputer erfolgt damit immer entsprechend dem vorhandenen Zeichenvorrat!
Beim Bereitstellen von Texten für die EDVA werden die Registerumschaltzeichen vom Emulationsprogramm entsprechend eingeblendet.

3.2.3. Organisation der Bildschirmarbeit

Das Programm EM62 ist für den Einsatz in Bürocomputern mit einem Bildschirm von 16 x 64 Zeichen vorgesehen.

Wirkung der Formatsteuerzeichen:

Zeichen	Codierung	Wirkung		Printer
			Display	
HT	H'Ø9'	Horizontal-tabulation	-Positionierung auf erstes Zeichen nach geschütztem Bereich, wenn zwischen aktueller Cursorposition und Zeichenende ein geschützter Bereich liegt -sonst auf Zeilenanfang der Folgezeile	keine
LF	H'ØA'	Neue Zeile	-Positionierung auf Zeilenanfang der Folgezeile -Liegt der Zeilenanfang der Folgezeile in einem geschützten Bereich, wird auf erstes Zeichen nach dem geschützten Bereich positioniert	Zeilenschaltung auf Anfang der Folgezeile

Zeichen	Codierung	Wirkung		
			Display	Printer
VT	H'ØB'	Vertikal-tabulation	-Positionierung auf aktuelle Cursorpos. der Folgezeile -Liegt die aktuelle Cursorpos. der Folgezeile in einem geschützten Bereich, wird auf erstes Zeichen nach dem geschützten Bereich positioniert	Zeilen-schaltung auf Folge-zeile
CR	H'ØD'	Carrige RETURN	-Positionieren Cursor auf Zeilenanfang der aktuellen Zeile -Liegt Zeilen-anfang der aktuellen Zeile in einem geschützten Bereich, wird auf erstes Zeichen nach dem geschützten Bereich positioniert	Positio-nieren auf An-fang der aktuel-len Zeile

3.2.3.1.
Bedienung

Die Beschreibung der Tastatur ist dem Systemhandbuch, Teil I, zu entnehmen.
Spezielle Bedienungen des Programms EM62 sind im Teil "Anleitung für den Bediener" aufgeführt.

3.2.3.2.
Feldkonzept der Eingabe

Bedienereingaben werden nach Bedienen der Starttaste ENTER/ET1 vom Emulationsprogramm aufbereitet und für die Komponenten des DFÜ-Anschlusses bereitgestellt, wenn die eingegebene Nachricht in die Marken # und ! eingerahmt ist.

Diese Marken können gesetzt werden:
- vom Bediener,
- durch eine Ausgabenachricht des Rechners.

Auf dem Bildschirm können gleichzeitig mehrere Felder definiert sein. Bei Bedienen der Starttaste ENTER/ET1 wird jeweils das Feld gelesen, dessen #-Marke sich vor der aktuellen Cursorposition befindet.
Bei den Kommandos H'2Ø' und H'28' werden # und ! nicht übertragen. Bei Kommando H'3Ø' werden sie übertragen.

3.2.3.3.
Geschützte Bereiche

In den Bürocomputern werden durch die Codezeichen
- H'1Ø' Anfang geschützter Bereich,
' H'1C' Ende geschützter Bereich
geschützte Bereiche definiert (s. Pkt. 3.2.1.).

Das Emulationsprogramm wandelt diese Zeichen vor der Ausgabe zur EDVA in die vom EC 8562/EC 8564 verwendete Zweizeichenfolgen

- ESC H H'1B48' Anfang geschützter Bereich,
- ESC I H'1B49' Ende geschützter Bereich.

Analog wird bei Eingabetexten von der EDVA die Zweizeichenfolge in die maschineninternen Codezeichen zur Kennzeichnung geschützter Bereiche gewandelt.
Die Codezeichen H'1Ø' und H'1C' können vom Bediener der Bürocomputer nicht gesetzt werden (Abweichung zu EC 8562)!

3.2.3.4. Registerumschaltung

Bei Einbeziehung der kyrillischen Buchstaben werden die Registerzustände LAT und RÜS entsprechend Pkt. 3.2.2.2.2. durch Einfügen der Registerumschaltzeichen SO bzw. SI definiert.
Diese Zeichen belegen keine Bildschirmpositionen und werden vom Emulationsprogramm entsprechend dem vorhandenen Zeichenvorrat behandelt.

3.2.3.5. Zeilen- und Spaltenadressierung

Bei den Kommandos H'28', H'5Ø' und H'58' muß die Bildschirmposition des ersten Zeichens des Textes entsprechend EC 8562 wie folgt fortlaufend codiert werden:

Zeilenadressen ADR3 H'3Ø'...
Spaltenadressen ADR4 H'2Ø'...

Das Emulationsprogramm wandelt diese Adressen in die maschineninterne Form bzw. umgekehrt fortlaufend wie folgt:

Zeilenadressen H'ØØ'...
Spaltenadressen H'ØØ'...

Anlage

Kommandovorrat

Codierung	Bezeichnung	Bedeutung hinsichtl. Anschlußverhalten	Bedeutung hinsichtlich Geräteverhalten
H'2Ø'	Lesen	Senden nach Eingabeschluß (Taste ENTER/ET1)	Das durch den Cursor gesetzte Eingabefeld wird übertragen. Nach der Übertragung wird der Cursor auf die nächste freie Position hinter "!" positioniert
H'28'	Lesen mit Positionsangabe	Senden nach Eingabeschluß (Taste ENTER/ET1) und um ADR3 und ADR4 erweitertes Nachrichtenformat	analog H'2Ø'
H'3Ø'	Lesen Puffer	Senden unbedingt	Der gesamte Bildschirminhalt wird übertragen.
H'4Ø'	Schreiben	Empfang	Darstellung ab aktueller Cursor- bzw. Druckposition*
H'5Ø'	Schreiben ab Zeile	Empfang, wobei Nachrichtenformat um ADR3 erweitert ist.	Darstellung ab durch ADR3 angewiesene Zeile
H'58'	Schreiben ab Position	Empfang, wobei Nachrichtenformat um ADR3 und ADR4 erweitert ist.	Darstellung ab durch ADR3 und ADR4 angewiesene Position.

Codierung	Bedeutung	Bedeutung hinsicht. Anschlußverhalten	Bedeutung hinsichtlich Geräteverhalten
H'6Ø'	Schreiben mit Löschen	Empfang	Löschen des Bildschirms und Darstellung ab Bildschirmempfang *
H'7Ø'	Schreiben mit Teil-löschen	Empfang	Löschen der ungeschützten Bereiche und Darstellung ab erster ungeschützter Bildschirmposition *

* Befindet sich der Cursor in einem mit # begonnenem Eingabefeld, wird das Kommando nicht ausgeführt!

Anwendungsbeschreibung

Programmbezeichnung: Logisches Kopieren von Diskettendateien
Kurzbezeichnung: COPD 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1. Zweckbestimmung des Programms

Das Dienstprogramm COPD 1520 (SIEX) (Logisches Kopieren von Diskettendateien) dient dem Kopieren von Anwenderdateien, unter Reorganisationskriterien von einer Quell- auf eine Zieldiskette (Dienst- und Servicefunktion).
Im Gegensatz zu physischen Kopieren mit COPD 1520 (SIEX) (1 : 1 - Kopieren; Anlegen von Sicherheitskopien) dient dieses Programm zum Kopieren von Dateien/Abschnitten von Dateien einer Arbeitsdiskette auf eine Zieldiskette.

Zweck:

- Archivierung von Anwenderdateien auf einer Archivdiskette,
- Rückspeicherung einer archivierten Datei auf eine Arbeitsdiskette,
- Zusammenstellen von unsortierten sequentiellen Dateien bzw. Dateiabschnitten zu einer neuen Datei,
- Herauslösen einer Datei von einer Arbeitsdiskette zur Weiterverarbeitung über Weitergabedatenträger.

Anwendungsbeschreibung

Programm: COPD 1520 (SIEX)

Kopierfunktionen:

- Kopieren aller Dateien (COPY ALL),
- Kopieren einer Datei (COPY FILE),
- Kopieren eines Abschnittes einer Datei (COPY PART).

Reorganisations-/Servicefunktionen:

- Verdichten der Anwenderdatei durch Auslassen gelöschter (markierter) Sätze,
- Erzeugung des Dateianfangs am Spuranfang zur Sicherung einer Forderung des Basisdatenaustauschniveaus (Weiterverarbeitung Dateien über andere Systeme!)

2. Anwendungsbedingungen

Das Kopierprogramm befindet sich für die Ausführung komplett im Speicher und benötigt eine RAM-Kapazität von 14 K Byte (einschließlich des 4 K Byte Kopierbereichs) sowie die geforderte Maschinenkonfiguration.

2.1. Quelldiskette

Die Quelldiskette enthält die zu kopierenden Anwenderdateien.
Datenformate der Anwenderdateien:

- Ungeblocktes Satzformat ohne Einschränkungen in bezug auf Satzlänge, außer beim abschnittsweisen Kopieren (s. 3. Anstrich).

- Geblocktes Satzformat, unter der Einschränkung
  . 2BL + RCS ≦ 4 K Byte,
    BL Blocklänge,
    RCS Satzlänge,
    beim Kopieren in eine Datei auf der Zieldiskette,
  . Die Funktion Kopieren ist nicht ausführbar.
- Bei Kopieren von Teildateien (Beginn- bis Endeschlüssel) ist RCS ≦ 4096 möglich. Blockung ist verboten.

2.2. Zieldiskette

- Bei COPY FILE / COPY PART:
  Leere (keine gültigen Dateikennsätze) oder teilweise mit Anwenderdateien belegte Diskette,
- Bei COPY ALL:
  Leere, oder mit Anwenderdateien belegte Diskette.
  Diese Kopierfunktion erzeugt selbständig eine leere Diskette.

2.3. Quell-/Zieldiskette

Die physischen Sektorformate (Sektorlängen) müssen für beide Disketten übereinstimmen (128, 256,512, 1024 Byte).

2.4. Dateiformen

- Sequentiell aufgebaute Dateien können zum Verdichten mit der Funktion "Keine Übernahme gelöschter (markierter) Sätze" kopiert werden.
- Direktzugriffsdateien (gestreut aufgebaute Dateien, indexierte Dateien) und Dateien für den Schlüssel indexierten Zugriff sind mit Übernahme markierter Sätze zu kopieren, um die logische Struktur der Datei zu wahren.

2.5. Technische Mittel/Konfiguration

Folgende Geräteausstattung ist erforderlich:

- RAM-Arbeitsspeicher ≥ 14 K Byte,
- Bildschirm 1024 oder 1920 Zeichen,
- Drucker wahlweise,
- 2 Diskettenlaufwerke.

3. Beschreibung von Aufgabe und Lösung

3.1. Beschreibung der Aufgabe

Das Programm hat die Aufgabe, die auf Diskette vorliegenden Anwenderdateien nach unter Pkt. 1 formulierten Bedingungen zu kopieren. Dabei sind die Kennsatzangaben der Dateien, außer den Dateizeigern, nicht zu verändern. Der Operator muß die gewünschte Kopieroperation über eine Auswahlentscheidung aufrufen können:

- Kopieren aller Dateien
- Kopieren einer Datei
- Kopieren eines Abschnitts einer Datei

) . Kopieren ohne Ändern Dateiname

) . Kopieren mit Änderung Dateiname

) . Kopieren in eine existierende Datei

Die Übernahme/Auslassung gelöschter Sätze muß steuerbar sein.

3.2. Lösungsbeschreibung

3.2.1. Allgemeines

Das Dienstprogramm COPD realisiert folgende Funktionen:

- **Kommunikation (Bedienerdialog: Bediener-COPD),**
- **Laufwerkszuweisung,**
- **Druckerzuweisung,**
- **Anzeige/Druck der Diskettenverwaltung (Verzeichnis),**
- **Kopieren aller Dateien,**
- **Kopieren einer Datei,**
- **Kopieren eines Abschnitts einer Datei.**

Mit Tastatureingabe der Steuerziffern oder Alphazeichen werden die zur Ablaufsteuerung benötigten Kommandos eingegeben. In einem nachfolgenden Submenü werden die Kommandoparemeter für die Kopieroperationen spezifiziert.

Die Dialogarbeit der Kommunikation erfolgt über den Bildschirm:

- Kommandozeile (3. Zeile),
- Fehlerzeile (5. Zeile),

- Kommando-Menü (6. - 15. Zeile),
- Statuszeile (16. Zeile).

3.2.2. Prinzipielle Abläufe

Vor dem Start des eigentlichen Kopiervorganges werden eine Reihe von Paremetern/Bedingungen geprüft (in Abhängigkeit von der angewählten Funktion):

- Ausreichend Platz auf der Diskette ab letzter Datei bis Dateiträgerende ?
- Übergehungsanzeiger gesetzt ?
- Übereinstimmungen der Satz-/Blockformate beim Kopieren in eine Datei ?
- Plausibilität der Schlüsselanfangsposition und Schlüssellänge in bezug auf Satzformat beim Kopieren eines Abschnittes einer Datei ?

Bei der Funktion "Kopieren aller Dateien" wird die Bestimmung des Endes der letzten Datei übergangen und automatisch der Wert Spur Ø1/Sektor Ø1 initialisiert.
Das Kopieren erfolgt bei ungeblockten/geblockten Satzformat über das Einlesen der Sektoren in einen 4 K-Pufferbereich und das anschließende Ausgeben auf die Zieldiskette.

Sonderfall:

Liegt
- geblocktes Satzformat,
- Kopieren einer Datei,
- Letzter Block der Zieldatei nicht vollständig gefüllt

vor, wird satzweise kopiert.

Folgende Fälle werden beim Kopieren gemeldet, falls nicht ausreichend Platz auf der Zieldiskette vorhanden ist:

Meldung	Behandlungsmöglichkeit
1. EOD < EOV, EOE > EOV Daten sind kopierbar, jedoch kein Platz für Gesamtdatei	Datei unter Modifizierung EOE: = EOV kopierbar
2. EOD = BOE Leerdatei	Bediener entscheidet Übernahme der Leerdatei (Kennsatzübernahme)
3. EOD = BOE, EOE > EOV Leerdatei, kopierbar mit modifizieren EOE	wie 1. und 2.
4. EOD, EOE > EOV Kein ausreichender Platz für die Daten der Datei	Kopierversuch möglich, wenn COPY ohne Übernahme markierter Sätze angewiesen war
5. Auf der Indexspur sind alle Dateikennsätze bereits vergeben (DESTINATION DISK IS FILLED)	keine
6. EOD > EOE Datei ist gefüllt bzw. EOE wird (ist) beim Kopieren in die Datei überschritten werden (worden)	Kopieren in diese Datei ist nicht möglich

Beim Kopieren in eine Datei mit der Meldung EOD > EOE können verschiedene Bedingungen vorliegen:
- Die Zieldatei ist tatsächlich gefüllt.

- Der Datenbestand von Quell- und Zieldatei überschreitet die Kapazität der Zieldatei. Da Übernahme markierter Sätze spezifiziert war, wird kein Kopierversuch aktiviert.
- Der Datenbestand von Quell- und Zieldatei überschreitet die Kapazität der Zieldatei. Da keine Übernahme markierter Sätze spezifiziert war, wurde automatisch der Kopierversuch gefahren.

3.3. Dialogarbeit

3.3.1. Tastatur

Die Bedienung erfolgt über die Eingabe eines Steuerzeichens (1-3, A-G). Die alphanumerischen Steuerzeichen sind beliebig (Groß-/Kleinbuchstabe) eingebbar.
Als Textendetaste wird die Taste ET1 (oder jede andere Starttaste außer ET2) verwendet. ET2 bewirkt Rückkehr in den Monitor oder Rückkehr in das Grundmenü.

3.3.2.1. Bildschirm

Es werden die Komponenten

- Kommandozeile,
- Fehlerzeile,
- Zusammenstellung der Anweisungen (Menü),
- Statuszeile

angezeigt.

3.3.3. Kommandozeile

Die Kommandozeile wird zur Anzeige der vom Bediener über die Tastatur eingegebenen Anweisungen verwendet.
Sie umfaßt maximal 63 Zeichen und ist durch Begrenzungszeilen eingerahmt.

3.3.4. Fehlerzeile

Fehler, die bei der Abarbeitung auftreten, werden in der Zeile nach der unteren Rahmenzeile angezeigt. Je nach Fehler wird nach CI-Tastenquittung in das Grundmenü zurückgegangen oder entsprechend Aufforderung im Zusammenstellungsteil weitergearbeitet (Betätigung ET1 zum Eingabeabschluß).

3.3.5. Zusammenstellung der Anweisungen

Im unteren Bildschirmteil werden die möglichen Anweisungen angezeigt. Im Grundzustand des Programms ist es das Basismenü.
Erfordert eine angewählte Funktion weitere Eingaben (Funktionen A,B,C), so wird eine spezielle Darstellung eingeblendet. Das Basismenü zeigt im linken Teil die Definitionsanweisungen (ASSIGN-STATEMENTS) und im rechten die Aktionsanweisungen (WORK-STATEMENTS) an.

3.3.6.
Statuszeile

Auf der letzten Bildschirmzeile werden die angewiesenen Definitionen und aktuelle Parameter angezeigt.

- Diskettenname und Laufwerkszuweisung für Quelldiskette,
- Sektorzähler Quelldiskette während Kopiervorgang,
- Sektorlänge Quelldiskette,
- Diskettenname und Laufwerkszuweisung für Zieldiskette,
- Sektorzähler Zieldiskette,
- Dateiname der aktuellen Quelldatei (während Kopiervorgang).

3.4.
Anweisungen

Nach der Eingabe der Steuerziffer des Steuerzeichens wird die zugehörige Anweisung in der Eingabezeile angezeigt, sofern sie eine zusätzliche Spezifikation benötigt.
Ist die vorzunehmende Spezifikation nicht im Grundmenü enthalten, wird die erforderliche Kommandostruktur im Zusammenstellungsteil angezeigt.

Die Definitionsanweisungen (ASSIGN-STATEMENTS) dienen der logischen Zuordnung der Laufwerke zum Programm, zur Festlegung des Datenstroms Quelle nach Ziel sowie der logischen Zuordnung des Druckers.

Die Aktionsanweisungen (WORK STATEMENTS) dienen

- zum Ausführen der Kopiervorgänge,
- zum Anzeigen der Diskettenverwaltungen (Verzeichnisse),
- zum Druck der Diskettenverwaltungen (Verzeichnisse).

4.
Ein- und Ausgabedaten

4.1.
Eingabedaten

- Operationseingaben zur Steuerung der Kopiervorgänge über Tastatur,
- Quelldiskette mit den zu kopierenden Anwenderdateien,
- Zieldiskette für die Übernahme der Daten.

4.2.
Ausgabedaten

- Kopierte Dateien auf der Zieldiskette,
- Als Druckliste Verzeichnis der Dateien auf Quell- und Zieldiskette sowie Protokoll der Kopieranweisungen,
- Bildschirmanzeige der Verzeichnisse der Dateien auf der Quell- und Zieldiskette.

Anwendungsbeschreibung

Programmbezeichnung: Sortierprogramm Floppy Disk
Kurzbezeichnung: SORD 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1. Zweckbestimmung des Programms

Mit SORD werden Dateien nach einem Sortiermerkmal sortiert. Die Dateien sind bzw. werden auf Floppy Disk gespeichert. Aus einer unsortierten Datei bzw. Teildatei wird eine sortierte Datei erzeugt. Die unsortierte Datei bleibt erhalten. Die Dateien können aus geblockten oder ungeblockten Sätzen bestehen. Die physische Satzlänge ist beliebig.
Das Sortiermerkmal kann aus maximal 16 Stellen bestehen (als vorzeichenlose ganze Zahl betrachtet), beliebig wählbar.

Der Vergleich zweier Sortiermerkmale kann dezimal (numerisch) oder binär (alphanumerisch) erfolgen.
Es kann aufsteigend oder absteigend sortiert werden.
Markierte Sätze werden nicht berücksichtigt.

2. Anwendungsbedingungen

Die sortierte Datei (Ausgabedatei) muß vor Programmbeginn generiert sein (Dienstprogramm FGEN).

Unsortierte und sortierte Datei müssen den gleichen Dateiaufbau besitzen (Blockung, logische und physische Satzlänge, Blocklänge).
Jede Datei muß sich vollständig auf einem Datenträger befinden (Mehrdatenträgerdateien ggf. mit FGEN in mehrere Eindatenträgerdateien teilen).
Der Datenbereich (Extent) der sortierten Datei muß so definiert sein, daß er alle sortierten Sätze aufnehmen kann.

Können wegen der Speichergröße der Anlage nicht alle Sätze sortiert werden, so kann eine zweite sortierte Datei erstellt werden. Beide Dateien kann man dann mit MIXD zu einer sortierten Datei mischen.

Technische Mittel

Speicherbedarf: Mindestens 8 K Byte Anwenderbereich RAM

Geräte

- Floppy-Disk: 1 oder 2 Laufwerke
- Bildschirm: 1024 Bytes oder
  1920 Bytes

3. Beschreibung von Aufgabe und Lösung

Mit dem Programm werden die unter (1.) genannten Funktionen realisiert. Die Funktionsauswahl erfolgt nach dem angezeigten Menü über die Steuerziffern 1 - 5 bzw. A und B.

1 Zuweisung unsortierte Datei (Name, Laufwerk),
2 Zuweisung sortierte Datei,
3 Festlegung Sortierfolge (aufsteigend, absteigend),
4 Zusammensetzung Sortiermerkmal (relative Satzpositionen),
5 Festlegung Vergleich (binär, dezimal),

A Start der Sortierung,
B Sprung in den Monitor.

Bevor mit A die Sortierung gestartet wird, müssen die Kommandos 1 - 5 ausgeführt sein (wird geprüft).

Danach erfolgt die Anzeige der
- Zahl der Sätze der unsortierten Datei,
- Zahl der in die unsortierte Datei entsprechend dem Datenbereich aufnehmbaren Sätze,
- Zahl der entsprechend dem verfügbaren RAM-Bereich sortierbaren Sätze.

Nach Eingabe der relativen Satzadresse der unsortierten Datei, ab der mit der Sortierung begonnen werden soll, erfolgt das sequentielle Lesen der in die Sortierung einbezogenen Sätze der unsortierten Datei. Gleichzeitig wird eine Tabelle im Speicher aufgebaut, die die Sortiermerkmale enthält. Für den dezimalen Vergleich erfolgt eine gepackte Speicherung, so daß weniger Speicherbereich benötigt wird bzw. mehr Sätze sortiert werden können als beim binären Vergleich.

Nach Ende der Eingabe erfolgt die Sortierung. Dabei erhält jedes Sortiermerkmal (log. Satz) einen Index. Entsprechend der Sortierfolge erfolgt das direkte Lesen des Satzes aus der unsortierten Datei und seine Ausgabe sequentiell in die sortierte Datei.
Vor einem erneuten Sortierlauf muß SORD wieder geladen werden, da Programmteile durch die Sortiermerkmale-Tabelle überschrieben wurden (Bedienerhinweis).

4.
Ein- und Ausgabedaten

Eingabedaten

- Programmparameter durch Tastatur,
- Datensätze von unsortierter Datei.

Ausgabedaten

- Anzeige Anzahl Sätze
  (unsortierte und sortierte Datei, sortierbare Sätze),
- Fehleranzeige,
- Datensätze in sortierter Datei.

Anwendungsbeschreibung

Programmbezeichnung Initialisieren Kassette mit Bandmerke / HDR
Kurzbezeichnung: CATM 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1.
Zweckbestimmung des Programms

Das Dienstprogramm CATM 1520 (SIEX) dient

- der Vorbereitung von 1/8" - Magnetbandkassetten zur Datenaufzeichnung (Initialisierung) und
- der Kontrolle, ob das richtige Band zur Initialisierung verwendet wird.

Initialisierung:

Für die Erstellung von Anwenderdateien im SIMPLE- oder BASIC-Niveau ist eine leere oder zu überschreibende Kassettenmagnetbandseite (Spur) mit einer Bandmarke am Bandanfang zu beschreiben.
Bei der Erarbeitung von Anwenderdateien im COMPACT-Niveau ist

- eine leere zu überschreibende Kassettenmagnetbandspur am Bandanfang oder
- eine mit COMPACT-Dateien belegte Kassettenmagnetbandspur am Datenende des Volumens (Datenende der Spur)

mit einer Leerdatei (COMPACT-Niveau) zu beschreiben.

Um bei der Dateneufzeichnung im COMPACT-Niveau die Datensicherheit von Kassetten gewährleisten zu können, ist das Aufzeichnen von Leerdateien mit vorgegebenem Dateinamen im Rahmen dieser Dienstfunktion notwendig.

Kontrolle:

Zur Ausführung der obengenannten Funktionen ist in der Regel eine Kontrolle des Bandinhalts notwendig.
Dies wird durch die Anzeigefunktion ermöglicht.
Der Bandinhalt wird dazu blockweise angezeigt. Datenblöcke werden im Hexadezimalcode und (wenn darstellbar) als alphanumerische Zeichen auf dem Bildschirm angezeigt. Bandmarken werden mit der Anschrift "TAPEMARK" gemeldet.
Da die beiden Spuren (A/B-Seite) getrennte Datenträgereinheiten (Volume) darstellen, ist für die Initialisierung bzw. Aufzeichnung der COMPACT-Leerdatei die Vorspulfunktion erforderlich.
Die Anzeigefunktion ist ebenfalls als Testhilfe für den Programmierer bei der Arbeit mit 1/8"-Kassettenmagnetband einsetzbar.

2. Anwendungsbedingungen

Folgende Konfiguration ist erforderlich:

- RAM-Arbeitsspeicher $\geq$ 3 K Byte,
- Bildschirm 1024 oder 1920 Zeichen,
- 1 Kassettenmagnetbandlaufwerk K 5200.

Für die Arbeit mit CATM ist eine leere oder zu überschreibende Kassette erforderlich.

3.
Beschreibung von Aufgabe und Lösung

3.1.
Beschreibung der Aufgabe

Das Dienstprogramm soll die Initialisierung von leeren oder zu überschreibenden Kassettenmagnetbandspuren mit einer Bandmarke oder einer COMPACT-Leerdatei auslösen. Zur Gewährleistung der Initialisierung des richtigen Kassettenmagnetbandes ist der Inhalt des Bandes über eine Anzeigefunktion zu kontrollieren, um ein irrtümliches Zerstören der Daten zu verhindern.

3.2.
Lösungsbeschreibung

Es werden nachfolgende Funktionen realisiert:

1 Zuweisung Kassettenmagnetbandlaufwerk,
2 Initialisieren Band mit einer Bandmarke am Bandanfang,
3 Blockweise Anzeige der auf dem Band aufgezeichneten Daten,
4 Vorwärtsspulen an das Bandende,
5 Aufzeichnen einer Leerdatei (COMPACT-Niveau) am Bandanfang oder am Datenende des Volumes,
6 Rückkehr in den Monitor.

3.3. Dialogarbeit

3.3.1. Tastatur

Die Bedienung erfolgt über die alphanumerische Tastatur. Jede Anweisung wird durch Eingabe eines Steuerzeichens (A-F) angewählt. Die alphanumerischen Eingaben können in Klein- oder Großbuchstaben vorgenommen werden.
Als Eingabeabschlußtaste wird die Taste ET1 verwendet. ET2 besitzt Abbruchwirkung (Abbruch bei den Funktionen (3) und (5). Alle anderen Starttasten bewirken in der Funktion (3) den Rückschritt, ansonsten wie ET1.

3.3.2. Bildschirm

3.3.2.1. Allgemeines

Der Bildschirm dient der Bedienerführung. Es werden angezeigt:
- Eingabezeile,
- Fehlerzeile,
- Zusammenstellungsteil,
- Statuszeile.

3.3.2.2. Eingabezeile

Die Eingabezeile wird zur Anzeige der vom Bediener eingegebenen Anweisungen verwendet. Sie ist durch Begrenzungsstriche oberhalb und unterhalb markiert.

3.3.2.3. Fehlerzeile

Die Fehlerzeile dient der Anzeige auftretender Fehler. Sie folgt unterhalb der eingegrenzten Eingabezeile.
Fehlermeldungen sind generell mit der CI-Taste zu quittieren.

3.3.2.4. Zusammenstellungsteil

Es werden die möglichen Anweisungen dargestellt.
Im Anzeigemodus Funktion (3) wird der Bereich des Zusammenstellungsteils zur Anzeige der Blockinhalte benutzt.

3.3.2.5. Statuszeile

In dieser Zeile (15. Zeile) wird die logische Gerätenummer (D/E) des zugewiesenen Kassettenmagnetbandlaufwerks und die Funktion Rückwärts-, Vorwärtsspulen sowie Suchen des Datenendes des Volumes im Aktionszeitraum angezeigt.

3.4. Beschreibung der Funktionen

3.4.1. Zuweisung Laufwerk (A ASSIGN)

Es wird ein Kassettenmagnetbandlaufwerk über eine Eingabe der logischen Geräteadresse (D/E) zugewiesen.

3.4.2. Initialisieren Band mit Bandmarke (B INITIALIZE VOLUME WITH TAPEMARK)

Am Bandanfang wird eine Bandmarke aufgezeichnet. Befindet sich das Band nicht am Anfang, wird automatisch zurückgespult.

3.4.3. Anzeige Blöcke (C DISPLAY BLOCKS)

Nach einem automatischen Rückspulen des Bandes an den Anfang - wenn erforderlich - wird der erste Block angezeigt.
Die nachfolgende Eingabe bewirkt bei

- ET1: nächster Block
- nnnn ET;: Überlaufen nnnn Datenblöcke und Anzeige des nachfolgenden (nnnn+1) Blockes (Bandmarken werden nicht gezählt; HDR zählen als Blöcke)
- ET2: Abbruch
- Jede andere Starttaste außer ET1 und ET2 bewirkt die Anzeige des vorhergehenden Blocks, es sei denn, eine Bandmarke oder der Bandanfang wird erreicht.

Es wird erkannt:

- Band leer,
- Band beginnt mit Bandmarke,
- Dateiende-Bandmarke (Bandmarke nach Datenblock),
- Datenende-Bandmarke (Bandmarke nach Dateiende-Bandmarke),
- Datenblock.

Datenblockanzeige:

Zur Anzeige auf dem Bildschirm gelangt der Inhalt eines Datenblocks in hexadezimaler (linke Bildschirmseite) und alphanumerischer Darstellung (rechte Bildschirmseite). Die maximale Anzeigekapazität beträgt 128 Bytes im Format 16 Zeichen x 8 Zeilen. Wird diese durch Blocklängen 128 Bytes überschritten, ist durch ET1 - Betätigung der Seitenwechsel (Bytes 129 ff. des Blocks) möglich. ET2 bewirkt den Abbruch der Funktion. Alle Codes außer den Codespalten 2 und 7 des Hexacodes sowie H'7F' werden als nicht darstellbare Zeichen mit "." angegeben.

3.4.4.
Vorspulen an das Bandende (D FORWARD TO END OF TAPE)

Es wird an das Bandende gespult, um nach Wenden der Kassette die andere Seite bearbeiten zu können.

3.4.5.
Aufzeichnen Leerdatei (E WRITE COMPACT-HDR)

Nach Anwahl wird zur Eingabe aufgefordert:

x, name,

wobei x = { B am Bandanfang
= am Datenende Volume

name: Dateiname mit maximal 8 Zeichen: 1. Zeichen Großbuchstabe; zulässige Zeichen: Großbuchstaben + Ziffern

Nicht korrekte Kommandos werden zur Korrektur zurückgewiesen. Wurde x = E gegeben, wird das Band auf Vorhandensein einer solchen Datei geprüft.
Anschließend sind Kassettennummer und Erstellungsdatum einzugeben.

3.4.6.
Rückkehr in den Monitor (F MONITOR)

Das Dienstprogramm wird verlassen.

3.5.
Fehlermeldungen

Entsprechend der allgemeinen Fehlerliste werden die Fehler zur Anzeige gebracht.
Der Fehler "NOTHING ON TAPE" - Band ist leer - ist ein CATM-spezifischer Fehler.
Bei Abarbeitungsbeginn jeder Funktion wird die Bereitschaft der Kassette geprüft und gegebenenfalls

"NOFOUND CASSETTE"

(keine Kassette bzw. Laufwerk nicht verriegelt) ausgegeben.
Die Kassette ist somit einzulegen und zu verriegeln.
Das Programm wird nach der Fehlerquittung (CI-Taste) fortgesetzt.

Weiterhin wird erkannt:
- Hardwarefehler,
- Lesefehler,
- Klarsichtband am Bandende erreicht,
- Kassette hat Schreibschutz.

Nach diesen Fehlern wird in das Menü gesprungen.
Befindet sich das Programm im Menü, ist das Laufwerk entriegelt, die Kassette kann entnommen werden. Das gilt auch für die korrekte Funktionsabarbeitung.

4.
Ein- und Ausgabedaten

Eingabedaten:

Tastatur:	Anweisungseingaben
Kassettenmagnetband:	Leeres bzw. mit Dateien belegtes 1/8"-Kassettenmagnetband

Ausgabedaten:

Bildschirm:	Blockinhalt Datenblock im Hexadezimalcode und als alphanumerische Zeichen bzw. Angabe "Bandmarke"
Kassettenmagnetband:	- Bandmarke am Bandanfang, - Leerdatei (COMPACT-Niveau) am Bandanfang oder am Datenende des Volumes.

Anwendungsbeschreibung

Programmbezeichnung: Funktionsprüfprogramm
Kurzbezeichnung: FUPR 1520 (SIEX)

Inhaltsverzeichnis

1. Zweckbestimmung des Programms
2. Anwendungsbedingungen
3. Beschreibung von Aufgabe und Lösung
4. Ein- und Ausgabedaten

1.
Zweckbestimmung des Programms

Das Programm FUPR erfüllt folgende Funktionen:

- Es prüft nach der Installation einer Anlage, ob alle Gerätefunktionen einwandfrei arbeiten. Bei dieser vom Techniker durchzuführenden Prüfung wird die Anlage in ihrer Gesamtheit oder auch baugruppenweise geprüft.
- Das Programm dient nach Reparaturen/planmäßigen Wartungsmaßnahmen dazu, die einwandfreie Funktion der Baugruppen bzw. der gesamten Anlage nachzuweisen.
  Diese Prüfung ist vom Techniker durchzuführen.
- Das Programm dient dem Operator des Anwenders dazu, Baugruppen bei Verdacht auf Fehlerfunktionen einzeln zu prüfen.
- Der Entwickler/Hersteller kann mit dem Programm Dauertests von Baugruppen durchführen.
- Mit FUPR können k e i n e Inbetriebnahmeprüfungen/Funktionskontrollen für die Datenfernübertragung vorgenommen werden.
  Dazu sind Programme erforderlich, die den als Gegenstellen funktionierenden Groß- bzw. Zentralrechnern angepaßt sind.

2.
Anwendungsbedingungen

Das Programm ist an keine bestimmte Anlagenkonfiguration gebunden. Durch Vorgabe einer Spezifikationsziffer kann es auf die jeweils vorliegende Anlagenausstattung abgestimmt werden (s. Bedienungsanweisung). Im RAM-Speicher ist ein Arbeitsbereich von mindestens 6 K Byte notwendig.

Das Laden der einzelnen Baugruppenprüfprogramme erfolgt nach Tastatureingabe des Programmnamens. Damit ist das Laden einzelner Baugruppenprüfprogramme möglich, aber auch das Zusammenstellen eines Programmzyklus durch das sequentielle Laden mehrerer Prüfprogramme.

Die Zeitdauer des Programmlaufs kann direkt durch die Eingabe der vorgesehenen Programmzeit oder indirekt durch Eingabe der Zyklenanzahl gewählt werden.

3.
Beschreibung von Aufgabe und Lösung

Zielstellung ist die Schaffung eines Prüfprogramms, womit die Baugruppen einer Anlage unabhängig von deren Konfiguration gezielt überprüft werden können.

Dazu wurde ein modulares Paket von Baugruppenprüfprogrammen geschaffen, die unter der Regie eines Leitprogramms geladen und aufgerufen werden. Leitprogramm und Programmoduln sind im Speicher verschiebbar ladbar. Die Generierung erfolgt durch das Ladeteil des Programms entsprechend der Anfangsadresse des Anwenderbereichs im Arbeitsspeicher bzw. der jeweiligen Lage des Moduls im Programmzyklus. Gleichzeitig erfolgt dadurch die Bindung an das Leitprogramm. Leitprogramm und Programmmoduln korrespondieren ständig miteinander.

Das Programm ist überwiegend aus Makrobefehlen aufgebaut.
Zur Optimierung von bestimmten Zeitabläufen und der Speicherausnutzung sind einige Programmteile in der Mikrobefehlssprache geschrieben.

4. Ein- und Ausgabedaten

4.1. Eingabedaten

Im Leitprogramm erfolgen die Tastatureingaben zur Bestimmung von Programmdauer, Anfangs- und Endezyklennummer.
Alle für die einzelnen Programmoduln notwendigen Tastatureingaben gehen aus den jeweiligen Bedienungsanleitungen hervor.

4.2. Ausgabedaten

Es erfolgt die laufende Anzeige des Namens für das gerade arbeitende Programm und der aktuellen Zykluszahl in der Systemzeile des Bildschirms. Am Programmende wird die Programmzeit sowie Summe der Modulzyklen sowie der evtl. vorhandenen Fehlerzähler für alle abgearbeiteten Programmoduln durch das spezifizierte Ausgabegerät "Drucker" oder "Bildschirm" ausgegeben.
Alle Rückmeldungen durch die einzelnen Baugruppen (z.B. Kontrollangaben, Fehlermeldungen) erfolgen ebenfalls über das vorgewählte Ausgabegerät und sind in den jeweiligen Bedienungsanleitungen beschrieben.